



# Nuvation Energy High-Voltage BMS Modules

## Product Manual

Document ID: NE-PM-001 | Revision: 1.6, 2023-07-26

BMS Software Version: Descartes Update 1  
© 2023 Nuvation Energy

# Table of Contents

1. Introduction	2
1.1. About this Manual	2
2. Battery Topology Terminology	3
3. System Overview	4
3.1. High-Voltage Stack Controller	6
3.2. High-Voltage Power Interface	7
3.3. Cell Interface	8
3.4. Features	9
4. Installation Instructions	11
4.1. Stack Controller Mechanical Installation	11
4.1.1. Dimensions and Weight	11
4.1.2. Installation Location and Position	11
4.1.3. Mounting Instructions	12
4.2. Stack Controller Electrical Connections	13
4.2.1. Getting Started	13
4.2.2. Connect Ethernet	13
4.2.3. Connect GPIO-In	14
4.2.4. Connect GPIO-Out	15
4.2.5. Connect CAN 485 (Optional)	16
4.3. Cell Interface Mechanical Installation	17
4.3.1. Dimensions and Weight	17
4.3.2. Installation Location and Position	18
4.3.3. Mounting Instructions	19
4.4. Cell Interface Electrical Connections	20
4.4.1. Getting Started	20
4.4.2. Connect Temperature Cable	21
4.4.3. Connect Battery Cells Cable	22
4.4.4. Connect the Link Bus Cable	28
4.5. Power Interface Mechanical Installation	29
4.5.1. Dimensions and Weight	29
4.5.2. Installation Location and Position	29
4.5.3. Mounting Instructions	30
4.6. Power Interface Electrical Connections	31
4.6.1. Getting Started	31
4.6.2. Connect Contactors	32
4.6.3. Connect Interlock	34
4.6.4. Connect Stack Bus	34
4.6.5. Connect Current Shunt	35
4.6.6. Thermistor Connector (optional)	36
4.6.7. Connect Stack Power	36
4.6.8. Connect External Power	37
4.7. First Power-up	38

4.7.1. Status LEDs	39
5. Setting up the Operator Interface	40
5.1. Download and Install the Operator Interface	40
5.2. Connect to the Operator Interface	41
5.3. Upgrading the Nuvation Energy BMS	44
5.3.1. Upgrading the Firmware	44
5.3.2. Upgrade Troubleshooting	45
5.4. Generate a Configuration File	46
5.4.1. Fine-tuning the Configuration File (optional)	46
5.5. Import the Configuration File	48
5.5.1. Configuration File Import Troubleshooting	49
5.6. Calibrate your Nuvation Energy BMS	50
6. Using the Operator Interface	51
6.1. The Dashboard Tab	51
6.1.1. Warnings and Faults	51
6.1.2. Stack Voltage	52
6.1.3. Stack Current	52
6.1.4. State-of-Charge	52
6.1.5. Depth-of-Discharge	53
6.1.6. Cell Voltage	53
6.1.7. Temperature	54
6.1.8. Nuvation Energy BMS Status	55
6.2. The Details Tab	58
6.2.1. Addressing	58
6.2.2. Battery	58
6.2.3. Safety	59
6.2.4. Cell Voltages	60
6.2.5. Thermistor Temperatures	61
6.2.6. Open Wire	61
6.2.7. SoC Calibration	63
6.3. The Menu Options	64
6.3.1. Registers	64
6.3.2. Service	64
6.3.3. Connection	67
6.3.4. Locking and Unlocking	67
6.3.5. About	68
6.4. The Status Banner	70
6.4.1. Service Lockout Indication	70
6.4.2. Self Check Indication	70
6.4.3. Communication Loss	70
6.5. Understanding the Service Lockout and Self Check	71
6.5.1. Entering Service Lockout	71
6.5.2. Exiting Service Lockout	72
6.5.3. Self Check	72

6.5.4. Self Check Troubleshooting . . . . .	73
7. Configuration Settings . . . . .	74
7.1. Background and Terminology . . . . .	74
7.1.1. Register Data Model . . . . .	74
7.1.2. Units . . . . .	77
7.1.3. Storage Types . . . . .	79
7.1.4. Configuration File . . . . .	79
7.2. Factory Lockdown . . . . .	80
7.2.1. Configuring Factory Registers . . . . .	80
7.2.2. Enabling Factory Lockdown . . . . .	80
7.2.3. Clearing Factory Lockdown . . . . .	80
7.2.4. Restricted Register Write Operations . . . . .	80
7.3. Battery Parameters . . . . .	83
7.3.1. Stack Capacity . . . . .	83
7.3.2. Stack Cycle Count . . . . .	83
7.3.3. State of Charge (SoC) . . . . .	84
7.3.4. Open Circuit Voltage Curve . . . . .	86
7.3.5. Cell Resistance Estimation . . . . .	86
7.3.6. Nameplate Information . . . . .	87
7.4. Stack Topology . . . . .	87
7.4.1. Cell Inputs . . . . .	88
7.4.2. Thermistor Inputs . . . . .	88
7.5. Operational Limits . . . . .	88
7.5.1. Triggers . . . . .	90
7.5.2. Cell Voltage Thresholds . . . . .	91
7.5.3. Thermistor Temperature Thresholds . . . . .	93
7.5.4. Stack Current Thresholds . . . . .	96
7.5.5. Stack Voltage Thresholds . . . . .	97
7.5.6. External Controller Heartbeat . . . . .	100
7.6. Control Settings . . . . .	100
7.6.1. Stack Switch Functions . . . . .	100
7.6.2. Stack Current Limits . . . . .	104
7.6.3. Passive Cell Balancing . . . . .	105
7.6.4. Breaker Interruption . . . . .	107
7.7. Input / Output Assignment . . . . .	108
7.7.1. Contactor Outputs . . . . .	108
7.7.2. General Purpose Digital Outputs . . . . .	109
7.7.3. General Purpose Digital Inputs . . . . .	110
7.8. Protocol Settings . . . . .	111
7.8.1. CAN Bus . . . . .	111
7.8.2. RS485 Modbus RTU . . . . .	119
7.9. Measurement Calibration . . . . .	119
7.9.1. Thermistor Calibration . . . . .	119
7.9.2. Stack Current Calibration . . . . .	121

7.9.3. Stack Voltage Calibration . . . . .	123
7.10. Hardware Settings . . . . .	124
7.10.1. Link Bus Scan Period . . . . .	125
7.10.2. Cell Voltage Low-Pass Filter . . . . .	126
7.10.3. Cell Open Wire Detection and Diagnostics . . . . .	126
7.10.4. Fault Pilot Signal . . . . .	128
7.10.5. Under-Voltage Shutdown . . . . .	128
7.10.6. Clock Verification . . . . .	128
7.10.7. Open Shunt Detection and Diagnostics . . . . .	129
7.10.8. Short Shunt Detection and Diagnostics . . . . .	129
7.10.9. Temperature Consistency Check . . . . .	131
7.10.10. Contactor Life Tracking . . . . .	133
7.11. Watchdogs and Freshness . . . . .	134
7.11.1. Freshness . . . . .	135
8. Communication Protocols . . . . .	137
8.1. Modbus Protocol Support . . . . .	137
8.1.1. Modbus RTU . . . . .	137
8.1.2. Modbus TCP . . . . .	137
8.1.3. Implemented MESA Models . . . . .	137
8.1.4. MESA Model Structure and Nomenclature . . . . .	139
8.1.5. Operational Cases for MESA . . . . .	140
8.1.6. Accessing MESA Models . . . . .	144
8.2. CAN Bus Protocol Support . . . . .	145
8.2.1. CAN Reporting and Commands . . . . .	145
8.2.2. Data Format . . . . .	145
8.2.3. Data Size . . . . .	146
9. External Interfaces . . . . .	147
9.1. High-Voltage Stack Controller . . . . .	147
9.1.1. Ethernet . . . . .	147
9.1.2. CAN 485 . . . . .	147
9.1.3. GPIO-Out . . . . .	148
9.1.4. GPIO-In . . . . .	149
9.1.5. Link Out . . . . .	150
9.1.6. Stack Bus . . . . .	151
9.2. High-Voltage Power Interface . . . . .	152
9.2.1. Contactors . . . . .	152
9.2.2. Interlock . . . . .	153
9.2.3. External Power . . . . .	153
9.2.4. Stack Bus . . . . .	154
9.2.5. Thermistor . . . . .	155
9.2.6. Current Shunt . . . . .	155
9.2.7. Stack Power . . . . .	156
9.2.8. Reset Push-Button . . . . .	157
9.3. Cell Interface . . . . .	158

9.3.1. Battery Cells . . . . .	158
9.3.2. Temperature Sensors . . . . .	161
9.3.3. Link In . . . . .	162
9.3.4. Link Out . . . . .	163
10. Troubleshooting . . . . .	165
10.1. Faults and Self Check Failures . . . . .	165
10.1.1. Cell Voltage Faults . . . . .	165
10.1.2. Stack Voltage Faults . . . . .	166
10.1.3. Thermal Faults . . . . .	166
10.1.4. Stack Current Faults . . . . .	167
10.1.5. Pre-charge Faults . . . . .	168
10.1.6. Contactor Faults . . . . .	168
10.1.7. Diagnostics Faults . . . . .	169
10.1.8. Breaker Faults . . . . .	171
10.1.9. Watchdog Faults . . . . .	171
10.1.10. Miscellaneous Faults . . . . .	173
10.2. Lost/Forgotten IP Address . . . . .	174
10.2.1. Wireshark (Windows/Linux) . . . . .	174
10.2.2. Netdiscover (Linux only) . . . . .	174
Appendix A: Operating Limits . . . . .	175
High-Voltage Stack Controller . . . . .	175
Electrical Characteristics . . . . .	175
Environmental Conditions . . . . .	177
Standards and Certifications . . . . .	177
High-Voltage Power Interface . . . . .	178
Electrical Characteristics . . . . .	178
Environmental Conditions . . . . .	179
Standards and Certifications . . . . .	180
Cell Interface . . . . .	181
Electrical Characteristics . . . . .	181
Environmental Conditions . . . . .	182
Standards and Certifications . . . . .	182
Maximum Stack Deployment . . . . .	183
Appendix B: Ordering Information . . . . .	185
High-Voltage Stack Controller . . . . .	185
High-Voltage Power Interface . . . . .	185
Cell Interface . . . . .	186
Appendix C: Changing the Nuvation Energy BMS Network Configuration . . . . .	188
Modifying the Network Settings . . . . .	188
Appendix D: Factory Reset . . . . .	189
Appendix E: Best Practices . . . . .	190
Grounding . . . . .	190
Excess Cable Management . . . . .	191
System Noise . . . . .	191

DC Filtering . . . . . 192

Stack Controller Grounding . . . . . 193

Link Bus Power . . . . . 194

Registers . . . . . 196

The content in this document must be followed in order to ensure safe operation of Nuvation Energy BMS.



Do **NOT** energize the system until all connections to the Power Interface and Cell Interface modules have been made.



The wiring of the battery cell voltage and temperature sensing should be verified **before** connecting to the Cell Interface modules. The temperature sensing wires must be isolated from the cell voltage sensing leads. Although the Cell Interface includes protective circuitry to make it more resilient to brief wiring errors, the same circuitry can result in the battery cells being slowly discharged. Over time, these wiring errors can cause damage to the Cell Interface and/or the cells.



Properly insulate or remove any unused wires. Unused wires can couple excessive system noise into Nuvation Energy BMS which can disrupt communication and lead to undesirable behaviors.



Insulated handling is required of any connector carrying potentials over 60 V DC relative to chassis.



Please be aware of high voltages present in your system and follow all necessary safety precautions.



Nuvation Energy BMS relies on your system charger to charge the battery cells; do not leave your charger off while Nuvation Energy BMS is powered from the stack for prolonged periods of time. Nuvation Energy BMS should be shut down when the system is in storage to minimize the drain on the cells.



The provided module enclosures are not fire enclosures.



## 1. Introduction

Thank you for choosing Nuvation Energy.

Nuvation Energy High-Voltage BMS is an enterprise-grade Battery Management System with features that extend battery life, ensure safety, provide data analytics, and enable remote management.

You can take advantage of the highly configurable browser-based user interface and custom-tune Nuvation Energy BMS to your specific target application.

### 1.1. About this Manual

This *Nuvation Energy High-Voltage BMS Modules: Product Manual* is a comprehensive manual, providing:

- Details about all the features offered by your Nuvation Energy High-Voltage BMS
- Mounting and wiring instructions to install this product safely
- Guidance on integrating the device into your Energy Storage System
- Guidance on operating the Nuvation Energy BMS Operator Interface

If there is a requirement for a Nuvation Energy High-Voltage BMS to complete a safety certification (such as to UL 1973) there are some additional constraints. These constraint is described in detail in the *Nuvation Energy BMS: Safety Manual* (available upon request).



This document applies to Nuvation Energy BMS Descartes Update 1 software release (Firmware version 4.109.1, Operator Interface version 0.57.1). Content may be inaccurate or incomplete for other versions.



We thrive on your feedback and what we build is driven by your input. Please submit support tickets to [support@nuvationenergy.com](mailto:support@nuvationenergy.com).

## 2. Battery Topology Terminology

Energy storage systems are hierarchical in nature. Nuvation Energy has adopted the following definitions for battery pack topology:

### Cell

A Cell is the smallest unit of energy storage distinguishable by the Battery Management System. One Cell, as defined from the perspective of the BMS, may actually consist of one or more electrochemical cells connected in parallel. This subtlety is reflected in the nomenclature for completeness. For example, a "1p" Cell refers to a single electrochemical cell, while a "2p" Cell refers to two electrochemical cells connected together in parallel. From the perspective of the BMS, these topologies appear identical except for the capacity of the Cells.

### Group

A Group is a set of Cells connected in series and managed together. For example, 12 "1p" Cells in series are referred to as a "12s1p" Group, while 16 "2p" Cells in series are referred to as a "16s2p" Group. Grouping of Cells is highly application-specific and is defined in how BMS hardware interfaces are physically wired up to Cells.

### Stack

A Stack is one or more Groups connected in series. For example, five "14s2p" Groups connected in series could be described as a "5g14s2p" Stack. However, it is far more common to describe it as a "70s2p" Stack.

### Bank

A Bank is one or more stacks connected in parallel. For example, three "70s2p" Stacks connected in parallel are referred to as a "3x70s2p" Bank.

### Pack

A Pack is one or more Banks connected in series.

## 3. System Overview

The Nuvation Energy High-Voltage BMS family includes several modules that operate together as a complete system. Available modules are listed below.

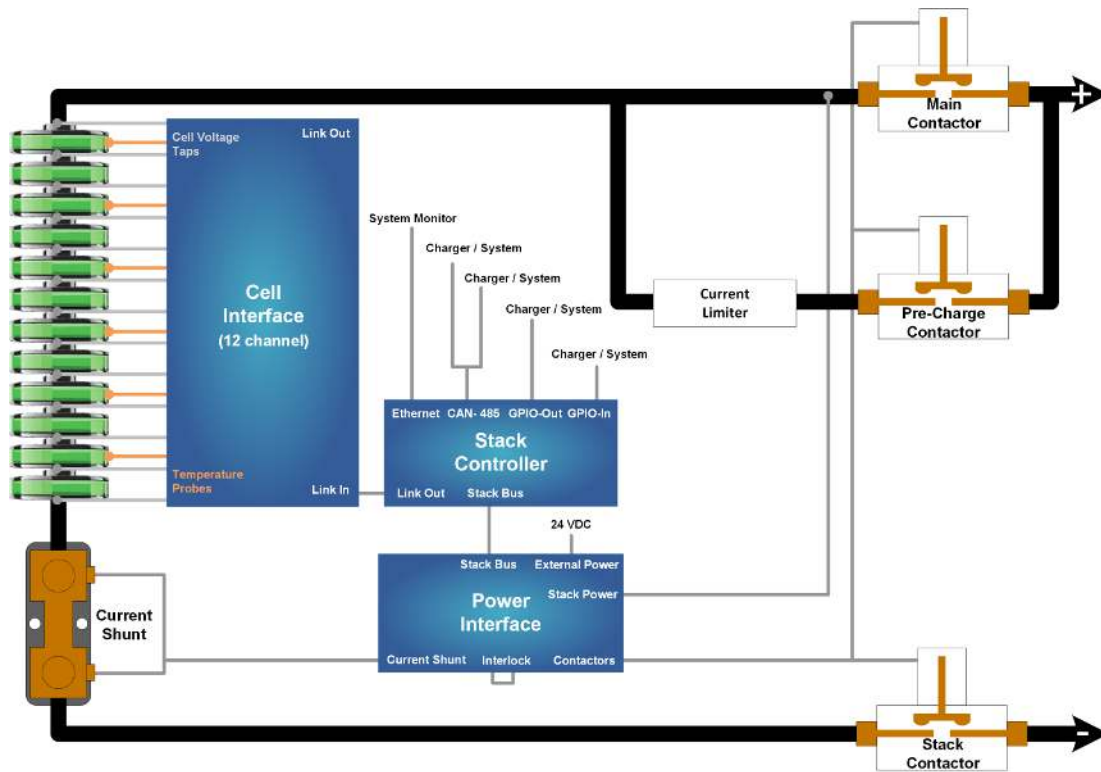
**Table 1. High-Voltage BMS Modules**

Model	Module Name
NUV100-SC	High-Voltage Stack Controller
NUV100-SC-NC	High-Voltage Stack Controller, no CAN
NUV100-PI-HE	High-Voltage Power Interface
NUV100-CI-12-1	Cell Interface - 12 channel
NUV100-CI-16-1	Cell Interface - 16 channel
NUV100-CI-4M12-1	Cell Interface - 12 V 4 channel



**Figure 1. Nuvation Energy High-Voltage BMS Modules**

Generally, a single High-Voltage BMS system uses one Stack Controller, one Power Interface, and one or more Cell Interface modules. An example configuration is shown in [Figure 2, “Nuvation Energy High-Voltage BMS System Overview”](#)



**Figure 2. Nuvation Energy High-Voltage BMS System Overview**

## 3.1. High-Voltage Stack Controller

The High-Voltage Stack Controller contains the central MCU which handles all the processes and decision making required by Nuvation Energy High-Voltage BMS. It monitors and controls all Cell Interface modules in a single battery stack.

There are two variants of the Stack Controller; the NUV100-SC — Stack Controller and the NUV100-SC-NC — Stack Controller, no CAN.



**Figure 3. Nuvation Energy High-Voltage Stack Controller Module**

The external interfaces to this module are:

- 10/100 Base-T Ethernet RJ45 jack (Modbus-TCP)
- Isolated CAN 2.0 port (not available in no-CAN variants (\*-NC))
- RS-485 (Modbus-RTU) connector
- 4 opto-isolated digital outputs
- 4 digital inputs
- Link Bus connector
- Stack Bus connector
- USB connector
- 3 Indicator LEDs

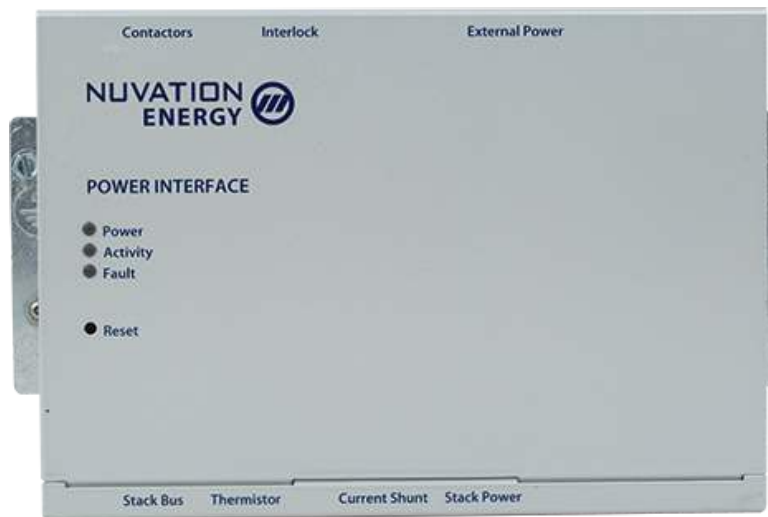
The Stack Controller does not have high-voltage connectors and does not connect to any battery stack referenced signals, making it safe to handle and connect to external equipment.

## 3.2. High-Voltage Power Interface

The High-Voltage Power Interface connects directly to high-voltage and high-current components. It accepts an external power input, provides power conditioning for all Nuvation Energy BMS modules and power for the contactors. The Stack Controller controls all operations on the Power Interface via the Stack Bus. The Power Interface contains a redundant MCU which handles all the processes and decision-making required by Nuvation Energy BMS to control the high-current contactors.

The Power Interface has high-voltage connectors and connects to battery stack-referenced signals. Safety precautions are required to handle and connect cables into this module.

There is only one model of the Power Interface, the NUV100-PI-HE.



**Figure 4. Nuvation Energy High-Voltage Power Interface Module**

The external interfaces to this module are:

- 4 high-current contactor coil drivers
- Interlock input
- External Power input
- Stack Voltage input
- Current Shunt input
- Stack Bus connector
- Thermistor input
- 3 Indicator LEDs
- Reset push-button

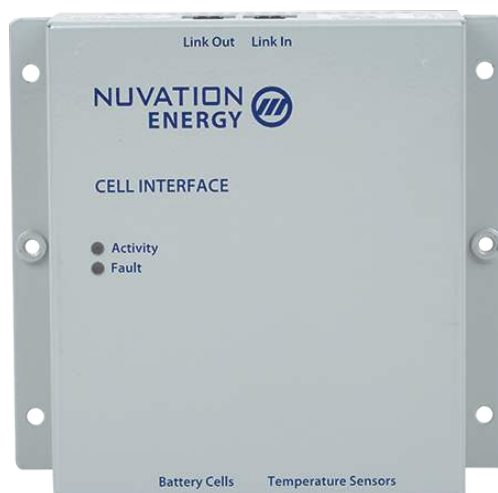
## 3.3. Cell Interface

The Nuvation Energy Cell Interface is the direct link between the individual battery stack cells and the rest of the Battery Management System. It facilitates battery monitoring and balancing functionalities.

In a High-Voltage BMS, one or more Cell Interface modules are used to convert and relay cell voltage and temperature measurements digitally to the Stack Controller. When using multiple Cell Interface modules, the same Cell Interface variant must be used—i.e. all NUV100-CI-12-1, or all NUV100-CI-16-1, or all NUV100-CI-4M12-1. The firmware does not support a mixed chain of different Cell Interface variants.

The following are variants of the Nuvation Energy Cell Interface:

- The NUV100-CI-12-1, Cell Interface - 12 channel can monitor up to 12 series-connected cells
- The NUV100-CI-16-1, Cell Interface - 16 channel can monitor up to 16 series-connected cells
- The NUV100-CI-4M12-1, Cell Interface - 12 V 4 channel can monitor up to 4 series-connected 12V lead-acid cells. Note that cell balancing is not supported in Cell Interface - 12 V 4 channel.



**Figure 5. Nuvation Energy Cell Interface Module**

The external interfaces to this module are:

- Battery cells connector
- Temperature sensors connector
- 2 Link Bus connectors
- 2 Indicator LEDs



The Cell Interface connects to the battery stack-referenced signals through high voltage rated connectors. Safety precautions are required to handle and connect cables into this module.

## 3.4. Features

The Nuvation Energy High-Voltage BMS modules that manage a battery stack include a Stack Controller, Power Interface, and Cell Interface. A single-stack system configuration includes one Stack Controller, one Power Interface, and one or more Cell Interface modules connected in series. This modular architecture supports battery pack voltages as high as 1250 V DC.

### Key software features

- Connects to Cell Interface modules to manage balancing and provide stack management.
- Calculates State of Charge and Depth of Discharge at both the cell and stack levels.
- Communicates with external systems via Modbus TCP (Ethernet), Modbus RTU (RS-485), or CAN bus.
- Provides access to the Nuvation Energy BMS Operator Interface.
- Supports firmware upgrades via Ethernet.

### RS-485

The Stack Controller provides an isolated RS-485 (Modbus) port. This interface can be used to control an external battery charger/inverter and for communicating with an external system (vehicle central controller, grid-attached controller, etc.)

### GPIO

The Stack Controller provides 4 isolated outputs and 4 general-purpose inputs to connect external systems or external equipment. The functionality of the GPIO interface is configurable by the end-user to match their needs.

### Link Bus

The Link Bus chain enables power and communication between the Stack Controller and the series-connected chain Cell Interface modules.

### Stack Bus

The Stack Bus provides a communication channel between the Power Interface and the Stack Controller. The Stack Bus also provides power from the Power Interface to the Stack Controller plus the summation of current consumed by all Cell Interface modules in the system.

### Contactors

The Power Interface module's Contactors connector enabled connecting to up to 4 external high-current contactor coils. The connector provides 24 VDC power for energizing contactor coils, and optional external coil power input for voltage flexibility.

### Interlock

The Power Interface module's Interlock connector provides a means to set the high-current contactor behavior. Using a physical switch/relay instead of a jumper is a convenient way to implement an interlock switch that de-energizes the system contactors.



The interlock feature can not be used for safety certified applications. For safety certified systems refer to [Section 9.2.1, "Contactors"](#) for details on how to implement a Vcoil interlock.



## **Current Shunt**

This interface is used to connect the Power Interface to a current shunt for stack charge and discharge current measurement as well as overall stack voltage measurement.

## **CAN**

An isolated CAN 2.0 port is available on the Stack Controller for communicating with external systems. This option is not available in no-CAN variants (\*-NC).

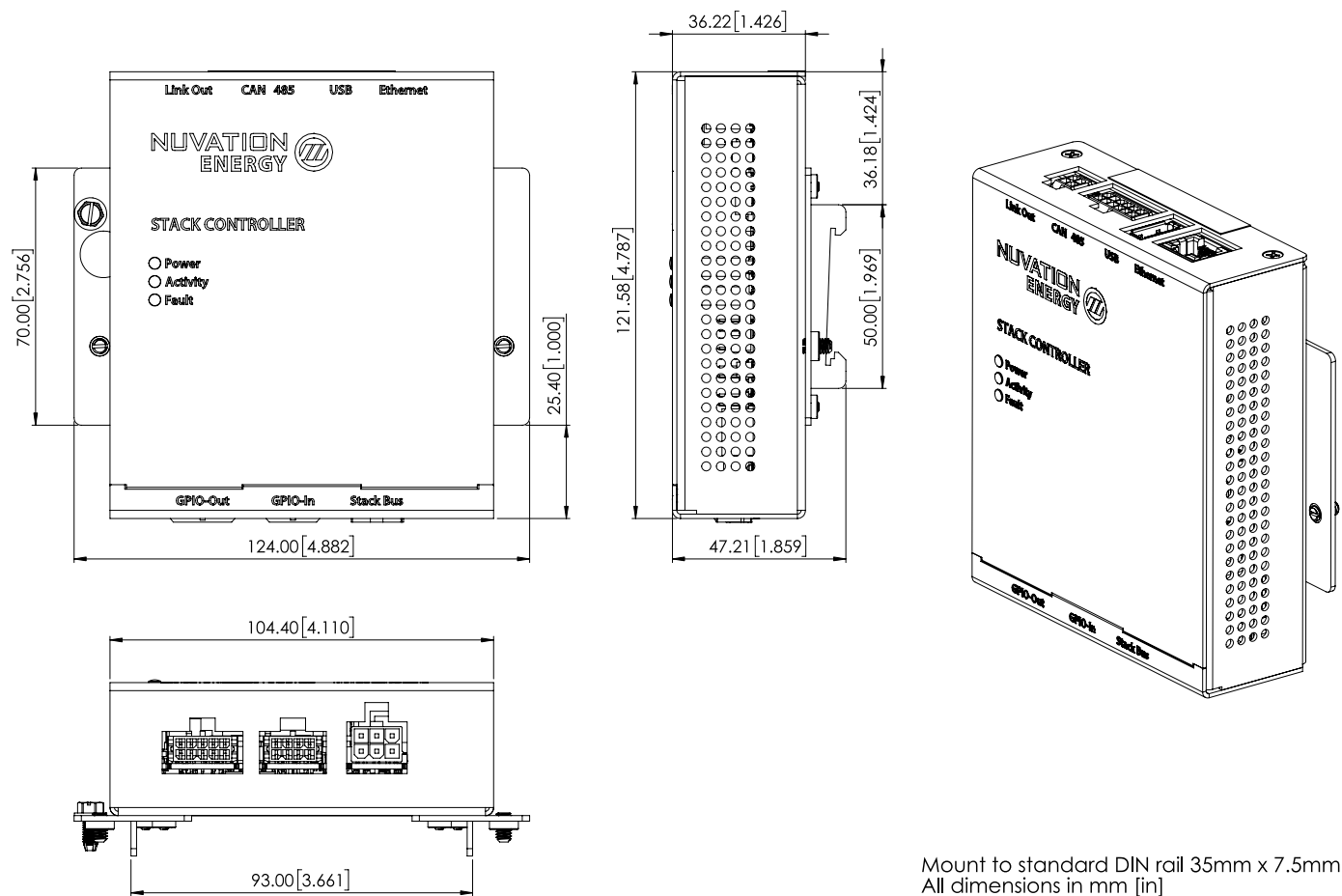
## 4. Installation Instructions

### 4.1. Stack Controller Mechanical Installation

#### 4.1.1. Dimensions and Weight

The overall dimensions of the Stack Controller are 104.4 mm × 121.58 mm × 40.6 mm.

The Stack Controller weighs approximately 525 g.



**Figure 6. Mechanical Drawing of Stack Controller**

#### 4.1.2. Installation Location and Position

The Stack Controller is rated to operate in the temperature range of -10 °C to 60 °C. It is designed for indoor use.

The Stack Controller is to be mounted to EN50022-compliant DIN rails.

##### 4.1.2.1. Mounting Clearances

Extra space should be provided around the module to allow for sufficient heat dissipation, and cable installation.

## 4.1.3. Mounting Instructions

### **DIN rail mounting**

Included with the Stack Controller are DIN clips that enable the Stack Controller to be securely mounted to EN50022-compliant DIN rails. There is a setscrew located on each DIN clip which tighten the clip's grip on the DIN rail to prevent the Stack Controller from sliding off.

The clips add an extra 19.6 mm to the overall width of the Stack Controller, bringing it from 104.4 mm to 124 mm. The clips also hold the module approximately 7 mm away from the inside lip of the DIN rail.

## 4.2. Stack Controller Electrical Connections

The Stack Controller does not have high-voltage connectors and does not connect to any battery stack-referenced signals, making it safe to handle and connect to external equipment.

### 4.2.1. Getting Started

To complete the installation of the Stack Controller, the following is required:

1. 1x Cable, CAT.5e, UTP, Ethernet
2. 2x Cable, GPIO, 300VDC (GPI and GPO)
3. 1x Cable, CAN 485, DB9 (Optional)

These cables are available for purchase as a kit from <https://nstore.nuvationenergy.com>.



**Figure 7. Example Stack Controller kit contents**

The instructions below are for applications using cable harnesses purchased from Nuvation Energy. To build custom cables, refer to [Section 9.1, “High-Voltage Stack Controller”](#) for connector pin-out and specifications.

#### Connection sequence

- Step 1: [Connect Ethernet](#)
- Step 2: [Connect GPIO-in cable](#)
- Step 3: [Connect GPIO-out cable](#)
- Step 4: [Connect CAN 485 cable](#) [Optional]

### 4.2.2. Connect Ethernet

This Ethernet (Modbus-TCP) interface is used as the primary means of connecting an external system to the BMS to configure the operating parameters and observe the status. It is also used as a means of communicating with the external system (vehicle central controller, grid-attached site controller, etc.)



**Figure 8. Ethernet cable (NUVW-D01X00-020)**

The Ethernet jack is a standard RJ45 Cat5e rated jack. The two LEDs on the Ethernet jack indicate link status (green LED) and network activity (yellow LED).

### 4.2.3. Connect GPIO-In

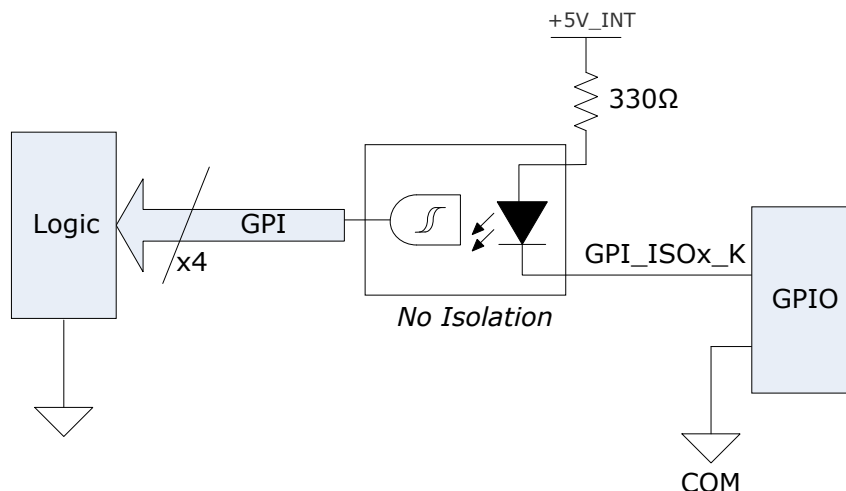
The GPIO-in cable is available with either 2 or 4 connections. Use the cable to connect external equipment to the GPIO-in interface.



**Figure 9. GPIO 4-in cable (NUVW-A06Z03-010)**

The functionality of this interface is configured by the end user to match their needs.

The GPIO-In connector provides 4 general-purpose inputs to allow an external system to send digital output to the Stack Controller. 4 independent detector circuits are used, driven by an on-board +5 V source. Each detector's input is connected to its corresponding pin, and paired with a COM reference pin per input. When the input is pulled to logic-low (connected to COM), each input will source about 12 mA to COM. Connect a dry-contact switch/relay or active-low/high-Z output capable of sinking up to 12 mA to use the input.



**Figure 10. GPIO Input High-Level Circuit Diagram**

## 4.2.4. Connect GPIO-Out

The GPIO-out cable is available with either 2 or 4 connections. Use the cable to connect external equipment to the GPIO-out interface.

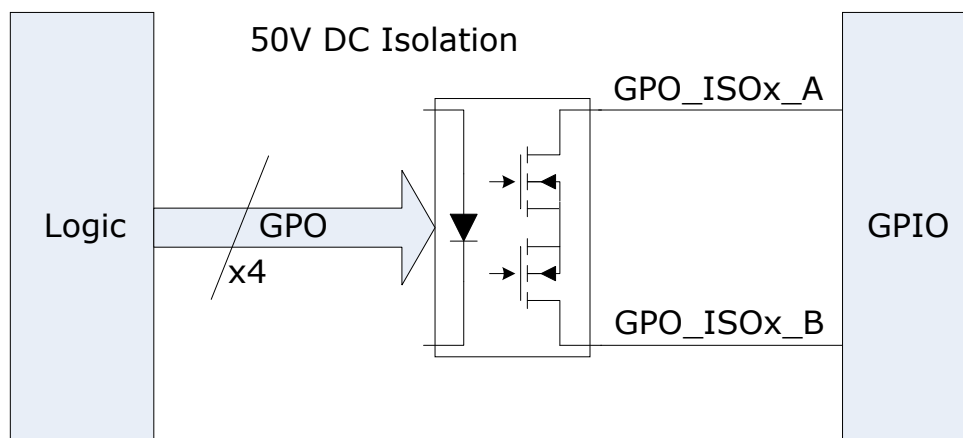


**Figure 11. GPIO 4-out cable (NUVW-A04Z03-010)**

The functionality of this interface is configured by the end-user to match their needs.

This interface provides 4 isolated outputs to allow an external system to receive digital input from the Stack Controller. 4 independent solid-state relays are used to connect \*\_A pins to their corresponding \*\_B pins.

Each output is rated for 60 V DC, 400 mA max., and the signals connected to each output must be within 50 V DC from chassis/earth ground. There is no polarity dependency between \*\_A and \*\_B pins.



**Figure 12. GPIO Output High-Level Circuit Diagram**

## 4.2.5. Connect CAN 485 (Optional)



The details below do not apply to no-CAN variants (\*-NC) as they do not have a CAN bus interface.

Connect the CAN 485 cable to external equipment. Refer to [Section 9.1.2, "CAN 485"](#) for connector pin-out.



This cable is sold in DB9 as well as RJ45 format.



**Figure 13. CAN 485, DB9 cable (NUVW-A07A03-010)**

This interface provides an isolated CAN 2.0 port and a non-isolated RS-485 (Modbus) port. It can be used to control an external battery charger/inverter and communicate with an external system (vehicle central controller, grid-attached controller, etc.)

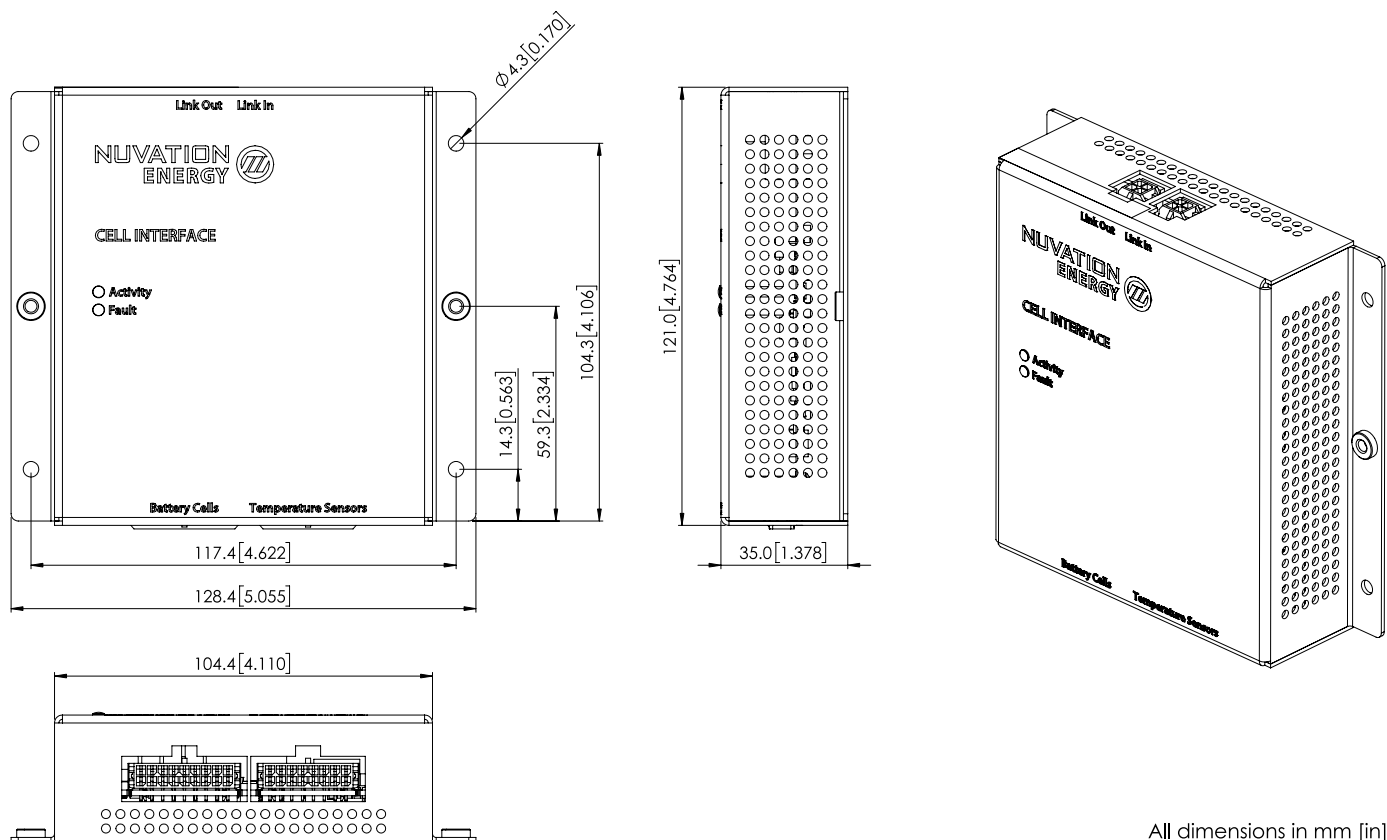
The isolated CAN 2.0 port requires 5.5 - 12 V DC sourced from an external power supply to operate.

## 4.3. Cell Interface Mechanical Installation

### 4.3.1. Dimensions and Weight

The overall dimensions of the Cell Interface are 104.4 mm × 121.58 mm × 40.6 mm.

The standard Cell Interface (i.e. with bulkhead) weighs approximately 450 g.



**Figure 14. Mechanical Drawing of Cell Interface with Bulkhead Enclosure**

#### 4.3.1.1. DIN rail mounting Kit

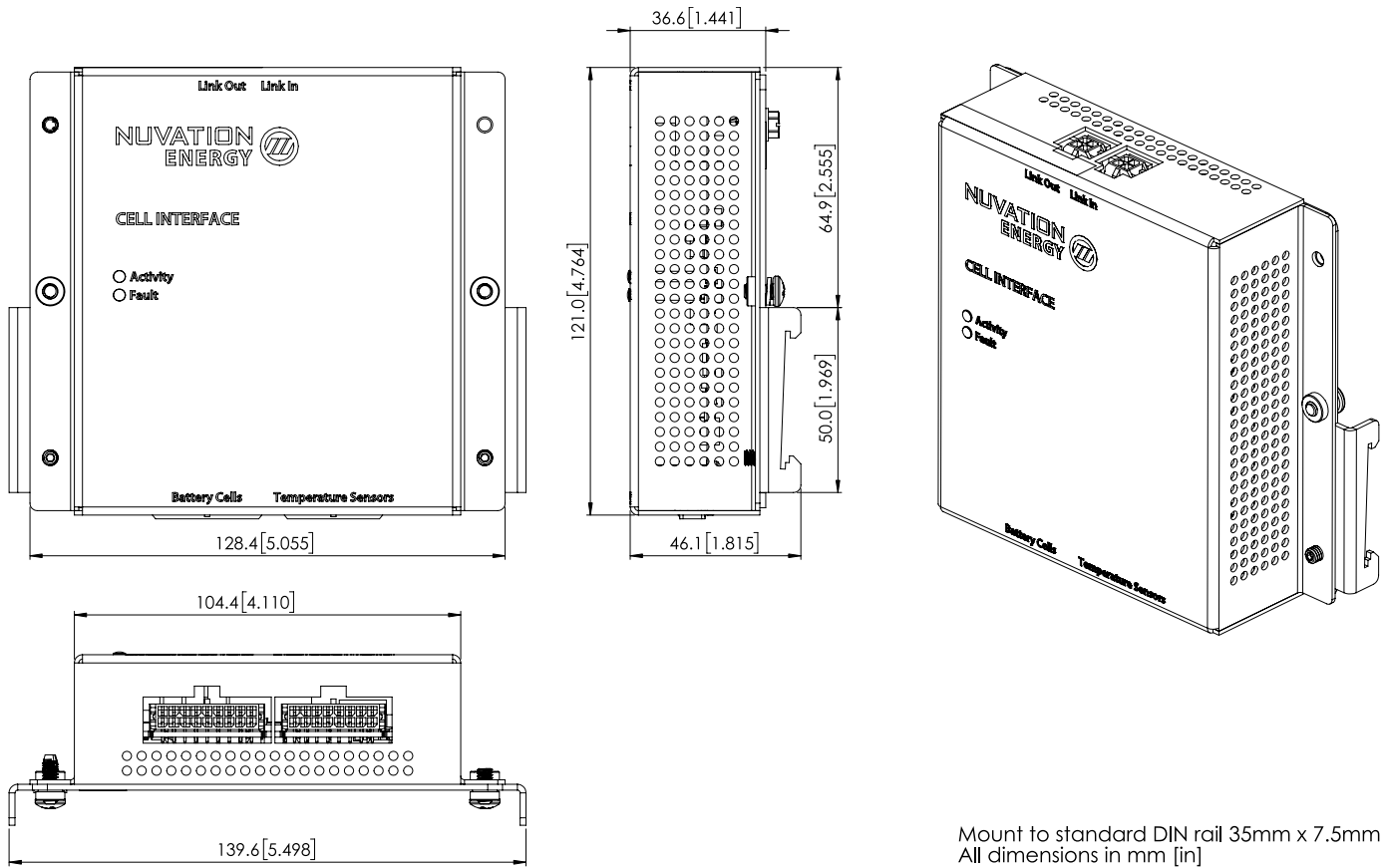
For applications requiring DIN rail mounting, the Cell Interface Mounting Bracket (Bulkhead-to-DIN) (sold separately) may be used. This kit includes a metal plate and the necessary hardware to securely mount the bulkhead enclosure of the Cell Interface to EN50022-compliant DIN rails.

The Mounting Bracket kit assembly adds an extra 14.2 mm to the overall width of the Cell Interface module, bringing it from 104.4 mm to 118.6 mm. The kit assembly holds the module approximately 7 mm away from the inside lip of the DIN rail.

The Mounting Bracket offsets the Cell Interface module from the center of the DIN rail approximately 30 mm upwards as shown in [Figure 15, "Mechanical Drawing of Cell Interface with Cell Interface Mounting Bracket \(Bulkhead-to-DIN\)"](#).

A Cell Interface with the Mounting Bracket weighs approximately 540 g.





**Figure 15. Mechanical Drawing of Cell Interface with Cell Interface Mounting Bracket (Bulkhead-to-DIN)**

## 4.3.2. Installation Location and Position



**Figure 16. Nuvation Energy Cell Interface Module**

The Cell Interface is available in a bulkhead-mountable enclosure which has five metal walls, leaving

the back of the unit fully exposed. It is designed to mount to a metal bulkhead panel such that the panel covers the exposed back.

The Cell Interface is rated to operate in the temperature range of -10 °C to 60 °C. It is designed for indoor use.

The Cell Interface should be mounted against a flat surface with the 'Link In' and 'Link Out' ports pointing up to the sky.



Ensure the enclosure is connected securely to Earth, either by grounding the mounting surface or by using a dedicated Earth Bonding Conductor. Refer to [Grounding](#) for details.

### 4.3.2.1. Mounting Clearances

The NUV100-CI-12-1 and NUV100-CI-16-1 variants produce up to 24 W and 32 W, respectively, during cell balancing. A portion of this heat is transferred to the bulkhead.

Extra space should be provided around the module to allow for sufficient heat dissipation, and cable installation.

If mounting a Cell Interface, PCB assembly only (no enclosure), note that the Cell Interface contains high-voltage signals reaching as high as 1250 V DC. Care must be taken when mounting the PCB into a metal enclosure to ensure that the metal walls remain a safe distance from the exposed conductor on the PCB. Using 1250 V DC as an example, the metal walls of the enclosure must be at least 4.2 mm from the nearest exposed conductor and must not touch the PCB or any component on the PCB, including the connector housings.

### 4.3.3. Mounting Instructions

Ensure appropriate fasteners (not included) are used to mount the Cell Interface. These fasteners will vary depending on the material and construction of the mounting surface.

It is suggested to use #8-32 screws with a split lock washer for the mounting holes provided on each side flange of the Cell Interface.

## 4.4. Cell Interface Electrical Connections



The Cell Interface connects to the battery stack-referenced signals through high-voltage rated connectors. Safety precautions are required to handle and connect cables into this module.

### 4.4.1. Getting Started

To complete the installation of the Cell Interface, the following is required per Cell Interface module:

1. 1x Battery Cells cable
2. 1x Thermistor cable
3. 1x Link Bus cable

These cables are available for purchase as a kit.



**Figure 17. Example Cell Interface kit (NUV100-CI-16-KIT-2)**

The instructions below are for applications using cable harnesses purchased from Nuvation Energy. To build custom cables, refer to [Section 9.3, "Cell Interface"](#) for connector pin-out and specifications.

#### Connection sequence

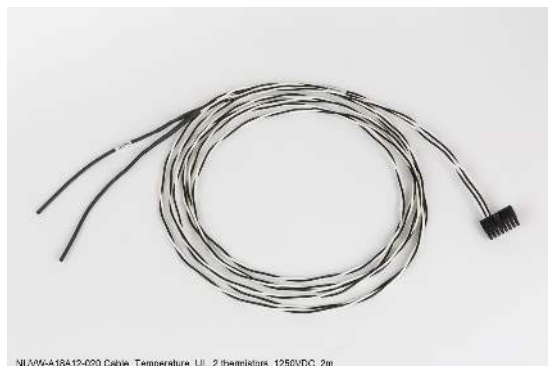
- Step 1: [Connect Temperature cable](#) for temperature measurement of the cells
- Step 2: [Connect Battery Cells cable](#) to battery voltage sense leads
- Step 3: [Connect Link Bus cable](#) to BMS module(s)



The wiring of the battery cell voltage and temperature sensing should be verified **before** connecting to the Cell Interface modules. The temperature sensing wires must be isolated from the cell voltage sensing leads. Although the Cell Interface includes protective circuitry to make it more resilient to brief wiring errors, the same circuitry can result in the battery cells being slowly discharged. Over time, these wiring errors can cause damage to the Cell Interface and/or the cells.

## 4.4.2. Connect Temperature Cable

The Temperature cable provides 10 kΩ NTC thermistors for temperature measurement of the cells and/or surrounding area. These sensors are used by Nuvation Energy BMS to detect over and under temperature conditions and adjust battery current limits to compensate for cell temperature.



**Figure 18. 2 Thermistor cable (NUVW-A18A12-020)**



**Figure 19. 7 Thermistor cable (NUVW-A19A12-020)**



2 to 7 thermistors are supported to accommodate systems requiring Nuvation Energy BMS to meet functional safety requirements.

### Cable Isolation

Because the sensors are referenced to the CELL0 or BLOCK0 input on the Cell Interface, care must be taken to ensure that they are electrically isolated from any common or ground potential, and from all other cell voltage terminals of all Cell Interface modules in the system

The thermistors must be isolated from the cell voltage terminals, as well as any metal work or other exposed conductors, in such a way that they will not make an electrical connection to a cell terminal in the event of vibration/failures.

### Thermal Consistency

For safety certified systems, there is an additional constraint on thermal consistency for all temperature measurements for each Cell Interface. The constraint is described in detail in [Section 7.10.9, "Temperature Consistency Check"](#).

## 4.4.2.1. Connection Procedure

### Connect to Surface

It is recommended to use thermally conductive/electrically non-conductive epoxy to adhere the thermistors to the cells.

### Connect to Cell Interface

'Temperature' cable connects to the 'Temperature Sensors' port at the bottom of the Cell Interface.

## 4.4.3. Connect Battery Cells Cable

The Battery Cells cable provides cell voltage input and a means for balancing the cells.

### Power Requirements

There is no minimum cell voltage requirement when powering the Cell Interface module from the Link Bus.

However, if powering the Cell Interface - 12 channel module from the cells (i.e. not powering from the Link Bus), a minimum of 11V must be present between the most negative and most positive cells.

Similarly, if powering the Cell Interface - 16 channel module or the Cell Interface - 12 V 4 channel module from the cells (i.e. not powering from the Link Bus), the cells are connected as two groups and a minimum of 11 V must be present between the most negative and most positive cells of each group.

## 4.4.3.1. Connection Procedure

### Connect to Cells

Connect the cables wires to the battery voltage sense leads. The individual wires are labelled to help with identifying the connections. Verify the cell voltages present on the connector are correct. While the Cell Interface module is tolerant of most wiring errors, incorrect connections may cause the cells to discharge.

Refer to the wiring examples below for the specific Cell Interface module used.

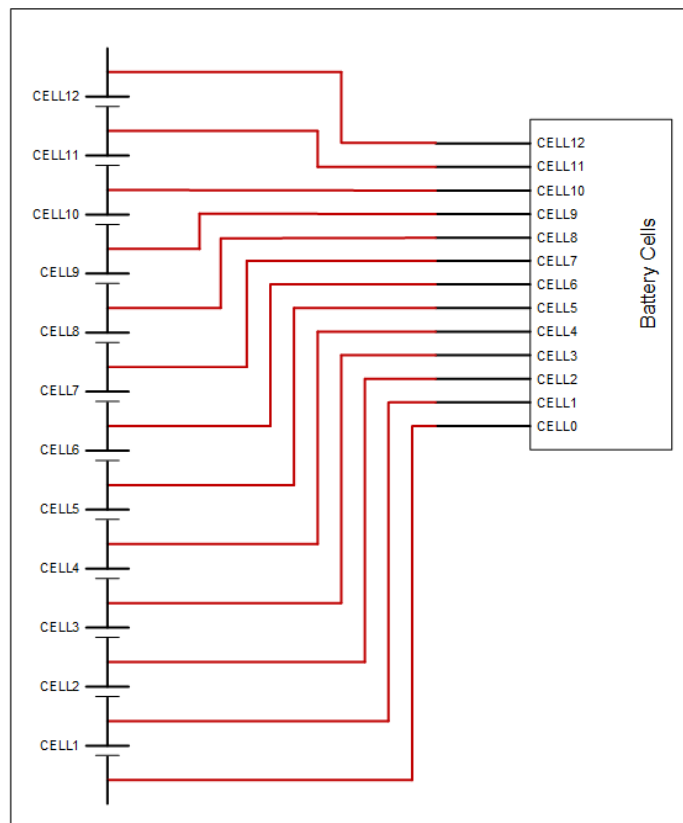
### Connect to Cell Interface

'Battery Cells' cable connects to the 'Battery Cells' port at the bottom of the Cell Interface.

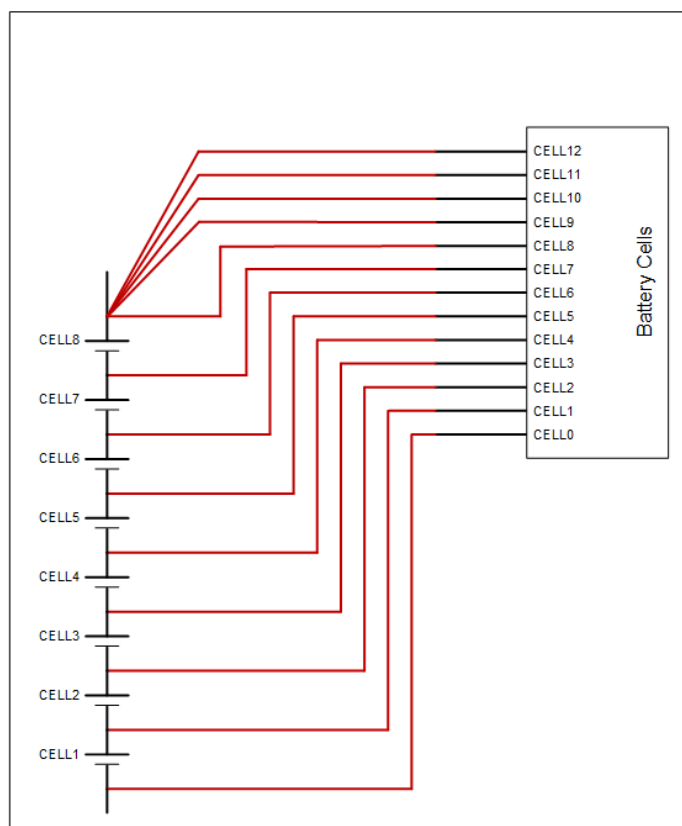
#### 4.4.3.1.1. Battery Cell Connector for Cell Interface - 12 channel



**Figure 20. Battery Cells cable (NUVW-A08Z12-020) for Cell Interface - 12 channel**



**Figure 21. Example wiring 12 cells in a Cell Interface - 12 channel**

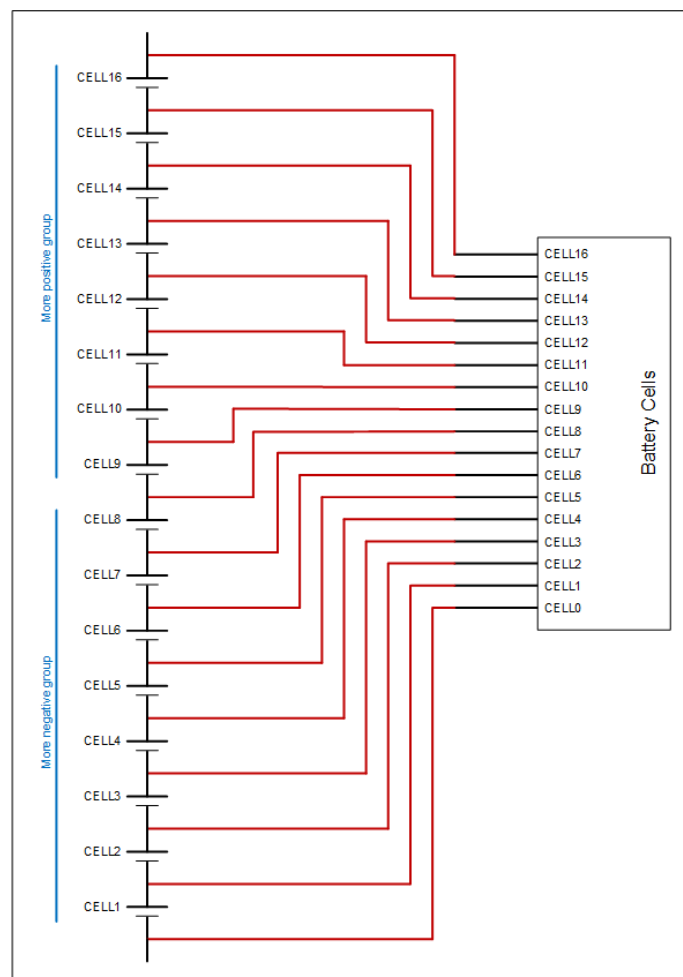


**Figure 22. Example wiring 8 cells in a Cell Interface - 12 channel**

## 4.4.3.1.2. Battery Cell Connector for Cell Interface - 16 channel

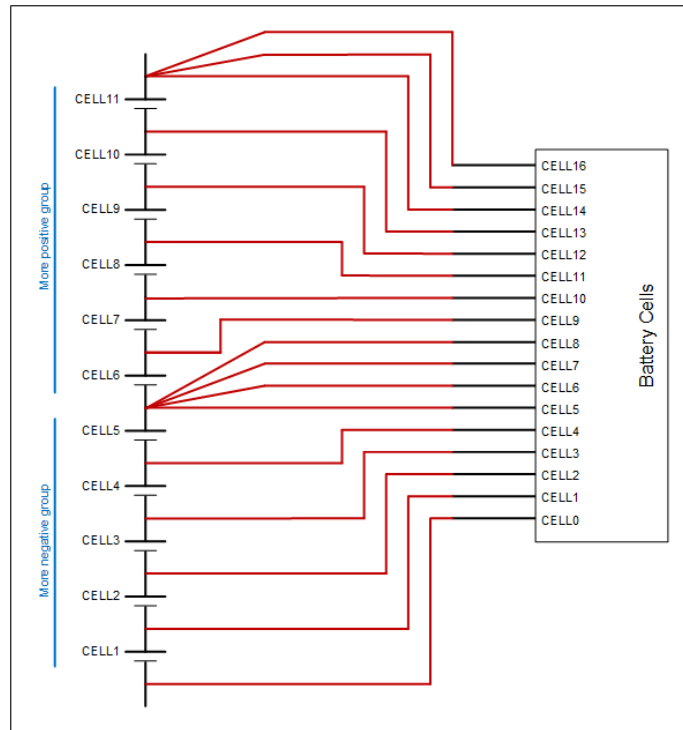


**Figure 23. Battery Cells cable (NUVW-A09Z12-020) for Cell Interface - 16 channel**



**Figure 24. Example wiring 16 cells in a Cell Interface - 16 channel**





**Figure 25. Example wiring 11 cells in a Cell Interface - 16 channel**

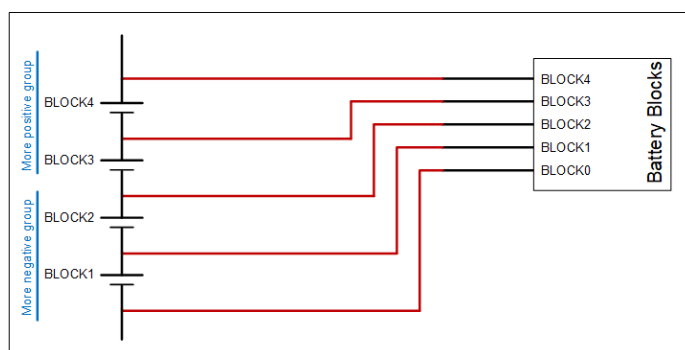
## 4.4.3.1.3. Battery Cell Connector for Cell Interface - 12 V 4 channel



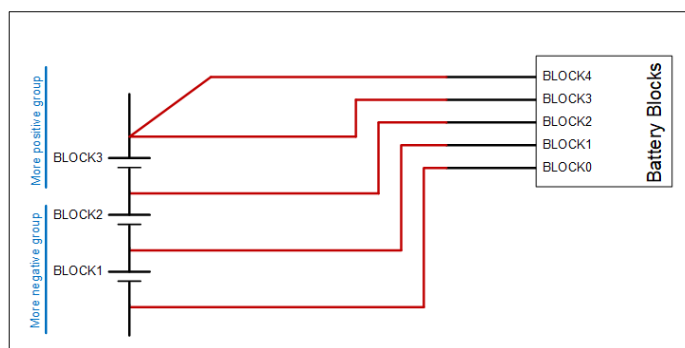
Balancing is not supported by the Cell Interface - 12 V 4 channel variant.



**Figure 26. Battery Cells cable (NUVW-A10Z12-020) for Cell Interface - 12 V 4 channel**



**Figure 27. Example wiring 4 blocks in a Cell Interface - 12 V 4 channel**



**Figure 28. Example wiring 3 blocks in a Cell Interface - 12 V 4 channel**

## 4.4.4. Connect the Link Bus Cable



**Figure 29. Link Bus cable (NUVW-A01X03-010)**

The Link Bus cable is used to connect Cell Interface modules in a daisy chain (called the Link Bus chain) by connecting the 'Link In' of one Cell Interface to the 'Link Out' of the next Cell Interface. In this series-connected chain of modules, each connection is made from the Link In connector of the module with higher potential to the Link Out connector of the module with lower potential.

The Link Bus provides a data channel and a power source to subsequent Cell Interface modules in the Link Bus chain. The amount of current supplied by this connector is the sum of current consumed by all subsequent Cell Interface modules in the Link Bus chain.

### Connection Procedure

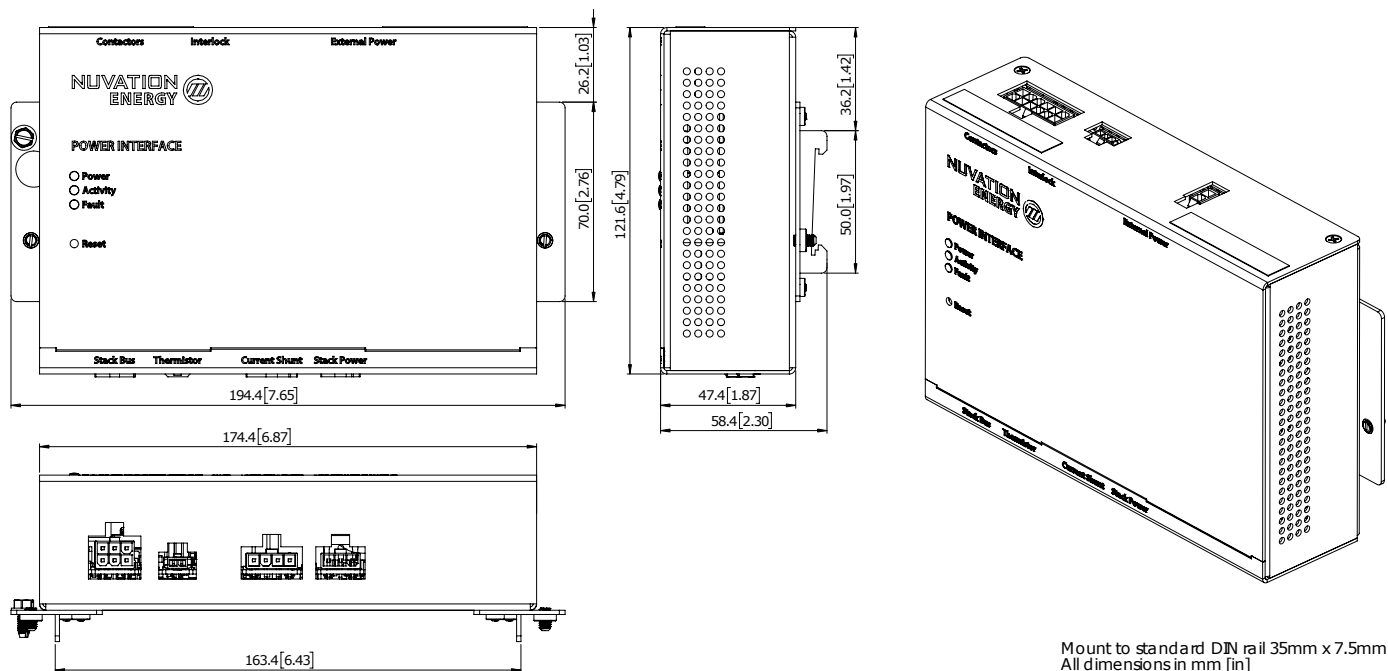
1. Connect one end of the cable to the 'Link Out' connector on the Stack Controller.
2. Connect the other end of the Link Bus cable to the 'Link In' connector on the Cell Interface.
3. For multiple Cell Interface modules, use additional Link Bus cables to connect the 'Link Out' on the current Cell Interface to the 'Link In' of the next Cell Interface in the chain.
4. The 'Link Out' of the last Cell Interface module in the Link Bus chain is to be left unconnected.

## 4.5. Power Interface Mechanical Installation

### 4.5.1. Dimensions and Weight

The overall dimensions of the Power Interface are 174.40 mm × 121.58 mm × 48.60 mm.

The Power Interface weighs approximately 915 g.



**Figure 30. Mechanical Drawing of Power Interface**

### 4.5.2. Installation Location and Position

The Power Interface is rated to operate in the temperature range of -10 °C to 60 °C. It is designed for indoor use.

The Power Interface is to be mounted to EN50022-compliant DIN rails.

#### 4.5.2.1. Mounting Clearances

Extra space should be provided around the module to allow for sufficient heat dissipation, and cable installation.

If mounting a Power Interface, PCB assembly only (no enclosure), note that the Power Interface can contain high-voltage signals. It is possible to have signals 1250 V DC away from earth ground. Care must be taken when mounting the PCB into a metal enclosure to ensure the metal walls remain the correct distance from the exposed conductor on the PCB. Using the 1250 V DC as an example, the metal walls must be at least 4.2 mm from the nearest exposed conductor and not touch the PCB or any component on the PCB, including the connector housings.

## 4.5.3. Mounting Instructions

### **DIN rail mounting**

Included with the Power Interface are DIN clips that enable the Power Interface to be securely mounted to EN50022-compliant DIN rails. There is a setscrew located on each DIN clip which tightens the clip's grip on the DIN rail to prevent the Power Interface from sliding off.

The clips add an extra 19.6 mm to the overall width of the Power Interface, bringing it from 174.40 mm to 194 mm. The clips also hold the module approximately 7 mm away from the inside lip of the DIN rail.

## 4.6. Power Interface Electrical Connections

### 4.6.1. Getting Started

To complete the installation of the Power Interface, the following is required:

1. 1x Cable, Stack Power, 1250VDC
2. 1x Cable, Current Shunt, 1250VDC
3. 1x Cable, Contactor, 300VDC
4. 1x Cable, Interlock, 300VDC
5. 1x Cable, External Power, 300VDC
6. 1x Part, Power Supply, 24VDC 2.5A, DIN Mount

These cables are available for purchase as a kit from <https://nstore.nuvationenergy.com>. The power supply is included in the base kit purchase.

Additionally, you will need the following items that are appropriately sized for your application:

1. 1x Current Shunt
2. 2x Contactors
3. 1x Pre-charge Contactor with current-limiting resistor [optional]

These items are available for purchase at <https://nstore.nuvationenergy.com>.



**Figure 31. Example Power Interface kit contents**

The instructions below are for applications using cable harnesses purchased from Nuvation Energy. To build custom cables, refer to [Section 9.2, “High-Voltage Power Interface”](#) for connector pin-out and specifications.

#### Connection sequence

- Step 1: [Connect Contactors](#)
- Step 2: [Connect Interlock Jumper](#)
- Step 3: [Connect Stack Bus](#)

- Step 4: [Connect Current Shunt](#)
- Step 5: [Connect Thermistor](#) (Optional)
- Step 6: [Connect Stack Power](#)
- Step 7: [Connect External Power](#)

## 4.6.2. Connect Contactors



**Figure 32. Contactor, 3-coil, UL cable (NUVW-B10Z03-010)**

The contactor interface is used to provide or select contactor coil operating power—either from an external power supply (40 V max) or from a loop-back connection from the BMS internal 24 V (nominal) source. The cable has the loop-back connection installed by default using orange wire for easy identification.

The Contactors connector connects to up to four external high-current contactor coils. Each output is capable of sourcing a maximum of 2.8 A continuously.

If coil operating power is provided from an external power source, the sum of all 4 output currents must not exceed 5 A continuous. If coil operating power is provided from the internal power source, the sum of all 4 output currents must not exceed 2.8 A or 2.9 A minus 31.7 mA per Cell Interface connected in the system, whichever is lower.

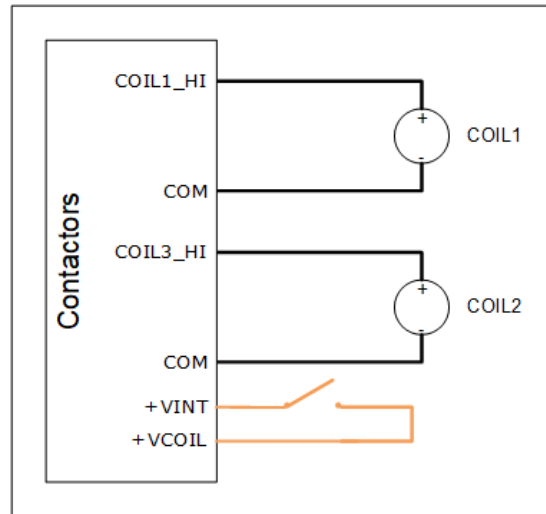
In safety certified applications that require a method to externally de-energize the contactors, a switch or relay can be used to interrupt the coil operating power for all contactors. Cut the orange loop-back connection and insert the switch or relay to be able to connect or disconnect the loop-back connection. The switch or relay must be rated for the anticipated worse-case current which will occur when all except one coil will be energized and then last coil pulls in its contact. The voltage rating of the switch or relay needs to be rated for 2x the coil voltage or up to 40V to survive transients.

In safety certified applications, two contactors should be used and one of the two contactors needs to be connected to Coil 1 or Coil 2 and the other contactor needs to be connected to Coil 3 or Coil 4. This implementation achieves redundancy against single point failures that would otherwise result in a situation where the battery cannot be disconnected from the external charging source or load.

Coil high-side drive and return outputs are provided at the connector. The return is referenced to the Power Interface chassis. Contactor coil back-EMF is internally clamped at 40 V.

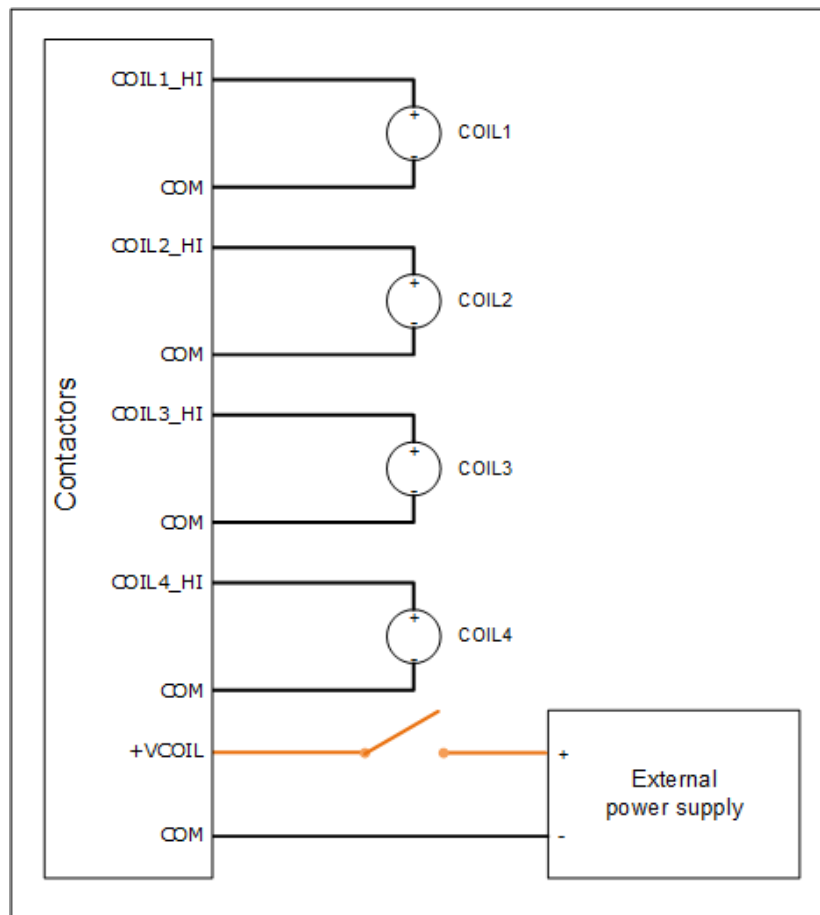
The following is an example 2 contactor implementation with 24 V coils that are powered from the

Power Interface:



**Figure 33. Example 2-coil wiring**

The following is an example 4 contactor implementation that is powered from an external power supply:



**Figure 34. Example externally powered 4-coil wiring**





The interlock switch shown in the Vcoil circuits for [2 coil](#) and [4 coil](#) contactors is required if an interlock functionality to externally de-energize the contactors is part of a safety certified system. In this case the switch used must be rated for the simultaneous power consumed by all contactors during pull-in. A non-safety certified system can omit the switch and implement an interlock directed by the interlock hardware functionality described in [Section 4.6.3, "Connect Interlock"](#).

## 4.6.3. Connect Interlock



**Figure 35. Interlock cable (NUVW-A02Y03-000)**

The Interlock connector provides a means to set the high-current contactor behavior, as outlined in the table below. Using a physical switch/relay instead of a jumper is a convenient way to implement an interlock switch that de-energizes the system contactors. It is recommended to connect pins 2 and 3 as this will enable the hardware redundant fault signaling feature to de-energize system contactors in the event of a Power Interface failure.



The interlock feature can not be used for safety certified applications. For safety certified systems, pins 2 and 3 must always be connected together. For systems that require an external interlock, refer to [Section 4.6.2, "Connect Contactors"](#) for details on how to implement a Vcoil interlock.

**Table 2. Interlock Options**

Pins 1 and 3	Pins 2 and 3	Function
open	open	System contactors are de-energized
open	closed	System contactors are controlled by Nuvation Energy BMS software, but de-energized if hardware-based fault signaling detects a fault
closed	either open or closed	System contactors are controlled by Nuvation Energy BMS software; hardware-based fault signaling mechanism will not de-energize system contactors



Pin 3 is electrically connected to chassis ground

## 4.6.4. Connect Stack Bus

Connect the Stack Bus cable to the Stack Bus connectors on the Power Interface and the Stack Controller modules.



**Figure 36. Stack Bus cable (NUVW-B01X03-010)**

The Stack Bus connector provides power and communication to the Stack Controller module.

The Stack Bus provides 42 mA to the Stack Controller plus the summation of current consumed by all Cell Interface modules in the system (up to 25 mA per CI-12 or 31 mA per CI-16/CI-4M12).

## 4.6.5. Connect Current Shunt

Connect the current shunt to this connector.



**Figure 37. Current Shunt cable (NUVW-B14Z12-010)**

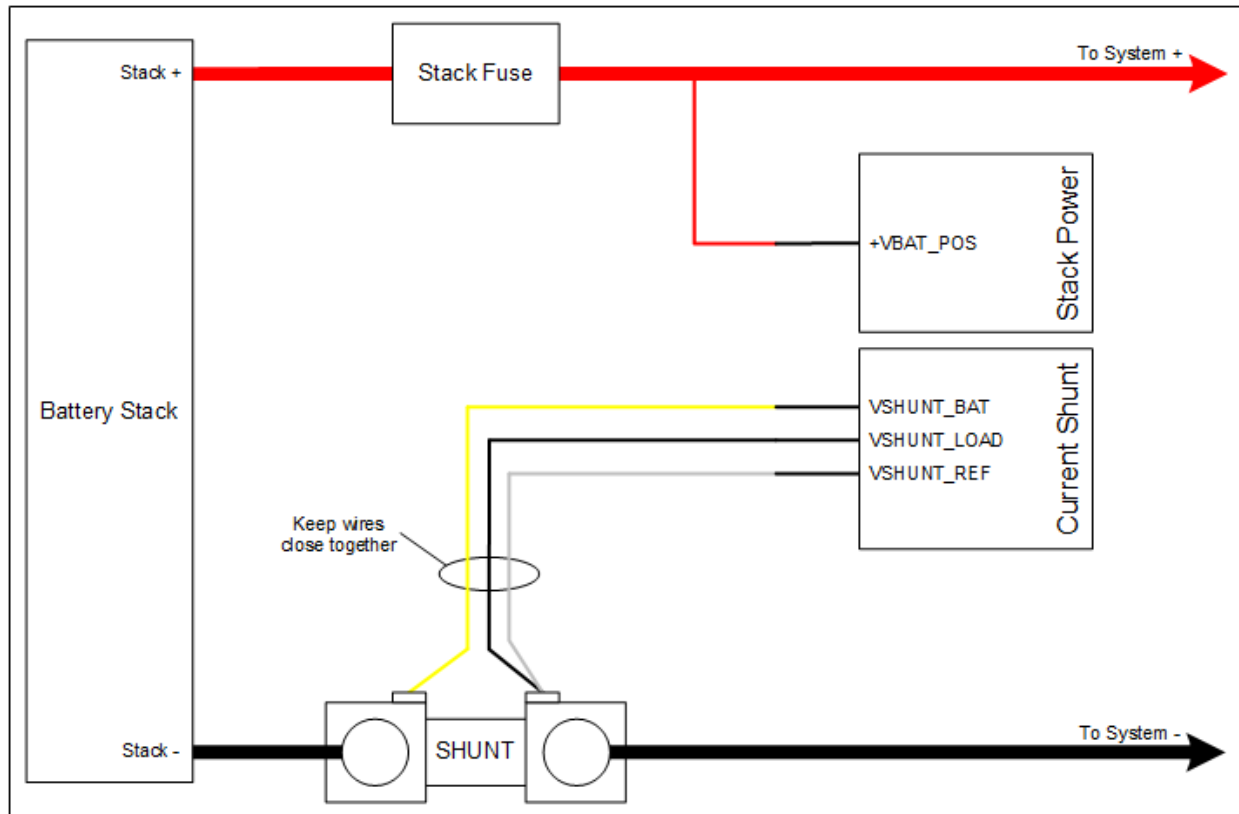
This interface is used to connect the Power Interface to a current shunt at the most negative end of the stack, for stack charge and discharge current measurement as well as overall stack voltage measurement.

The differential voltage across the shunt must never exceed 1V under any circumstance. Choose the resistance according to the application.



Full stack potential exists between the 'Stack Power' and the 'Current Shunt' connectors. Potentially dangerous voltages may also be present between either or both of these connectors and earth or chassis ground, including the Power Interface and Stack Controller enclosures. Appropriate safety precautions must be observed.

An example high-current shunt wiring diagram is shown below:



## 4.6.6. Thermistor Connector (optional)

The thermistor must be electrically isolated from the battery stack. Contact Nuvation Energy for support if temperature compensation of the high-current shunt is desired for your specific application.



Nuvation Energy can be contacted via [support@nuvationenergy.com](mailto:support@nuvationenergy.com).

## 4.6.7. Connect Stack Power



**Figure 38. Stack Power cable (NUVW-B02Z12-020)**

The Stack Power connector is used to provide an overall stack voltage measurement.

Connect the overall battery stack positive terminal to this connector.

## 4.6.8. Connect External Power

Included in the kit is DIN mountable, 5 A, 24 V DC, Power Supply. Connect the External power cable to the power supply.



**Figure 39. External Power cable (NUVW-B03A03-010)**



**Figure 40. Power Supply, 24VDC, 5A, DIN Mount (NUVX-OM-S8VK-C12024)**

The External Power connector accepts power from an external power supply. This is the sole power source for Nuvation Energy BMS.

If using another external supply, it can be either 16--24 V AC or 13-34 V DC and must be isolated from the battery stack, chassis, and COM grounds.

## 4.7. First Power-up

Once the High-Voltage BMS is connected to the battery (including the Cell Interface modules chain), the Power Interface can be powered. The Power Interface power LED (blue) should turn ON.

On initial power-up, it is common to see that the fault LED (red) is also ON. This indicates that a fault has been detected.

The High-Voltage BMS can now be connected to a network, either through a network switch or directly to a computer/laptop's network adapter. The default IP address of the High-Voltage BMS is 192.168.1.21. Refer to the subsequent [Section 5, "Setting up the Operator Interface"](#) and [Section 6, "Using the Operator Interface"](#) for instructions on setup and using the Operator Interface.

## 4.7.1. Status LEDs

When the Battery Management System is powered up, the status LEDs on the modules provide indication of the functional status of the module. A more detailed status may be accessed via the Operator Interface.

### 4.7.1.1. Stack Controller

The 3 LEDs are used by the Stack Controller to indicate health and functional status to the user:

- The Power LED indicates that the Stack Controller is operational
- The Activity LED indicates that the Stack Controller is processing data
- The Fault LED indicates that the Nuvation Energy BMS has detected a fault

### 4.7.1.2. Power Interface

The 3 LEDs are used by the Power Interface to indicate health and functional status to the user:

- The Power LED indicates that the Power Interface is operational
- The Activity LED indicates that the Power Interface is processing data
- The Fault LED indicates that the Nuvation Energy BMS has detected a fault

### 4.7.1.3. Cell Interface

The 2 status LEDs on the face of the Cell Interface indicate the functional status of the module.

#### **Activity**

This LED indicates that the Cell Interface module has received a communication packet over the Link Bus.

#### **Fault**

This LED is not used in normal operations. It is available to support on-site troubleshooting.

## 5. Setting up the Operator Interface

### 5.1. Download and Install the Operator Interface

The Operator Interface is used to access the High-Voltage BMS.

The latest Operator Interface release is available online at: <https://ncloud.nuvationenergy.com>. You may create an account to download the software package.

1. Download the High-Voltage BMS package
2. Extract the contents of the package to a suitable location on your computer.
3. Releases follow a naming convention similar to `nuvation-bms-0I-descartes.zip`. The package should be extracted to a folder with the same name to avoid overwriting other releases or files present in the same directory.



#### Browser Compatibility

The Operator Interface currently supports the most recent versions of Mozilla Firefox and Google Chrome. Other browsers such as Internet Explorer are not supported. Please install a supported browser before attempting to access the Operator Interface.

## 5.2. Connect to the Operator Interface

Ensure your computer is connected directly to the Stack Controller via an Ethernet cable. You will need to configure the network adapter on your computer to match the settings on your Stack Controller.

### Identify the Battery Management System IP

By default, Stack Controller is configured with a static IP address of 192.168.1.21.

### Identify the parts of the Battery Management System's IP Address

The IP address has two parts - the network ID and the host ID.



**Figure 41. Parts of an IP address**

Write down the network ID and host ID parts of the Stack Controller's IP address.

### Determine a valid IP address to assign to the PC

The Network ID for the PC IP address must be the same as the Stack Controller. The Host ID number can be any number from 2 to 255 as long as it is not the same as the Host ID of the Stack Controller. For example if the Stack Controller IP address is 192.168.1.21, the IP address of the PC could be 192.168.1.12 or 192.168.1.49 or any 192.168.1.x where x is not 21.



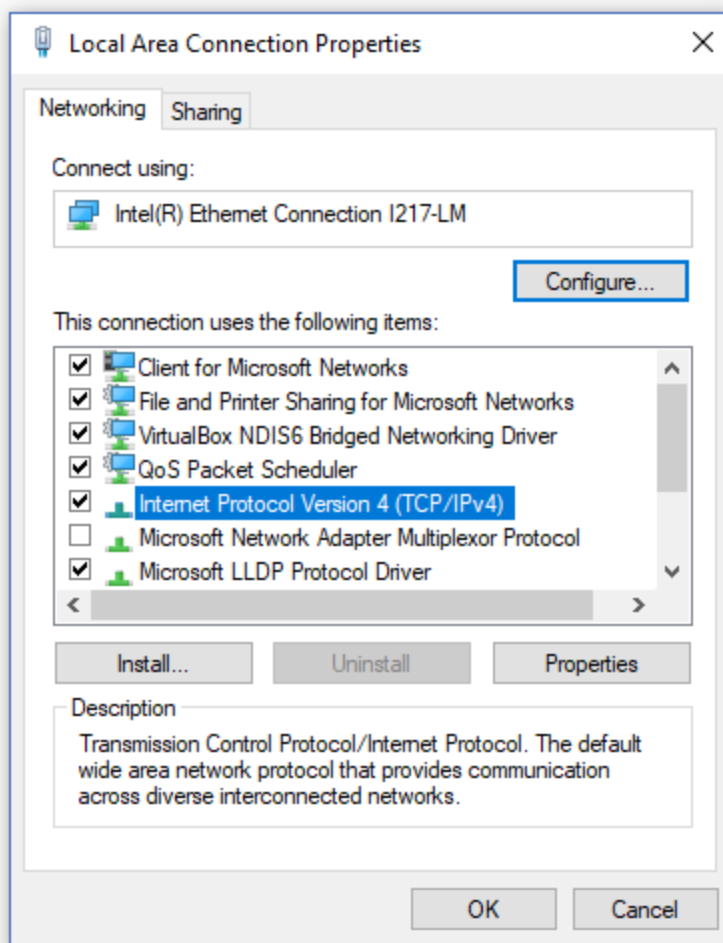
If you have multiple stacks, note down the Host ID from the IP address of each stack's Stack Controller. The Host ID number assigned to the PC must be different from this list.

### Setup the Static IP address on the PC

Instructions below are for a PC running Windows OS:

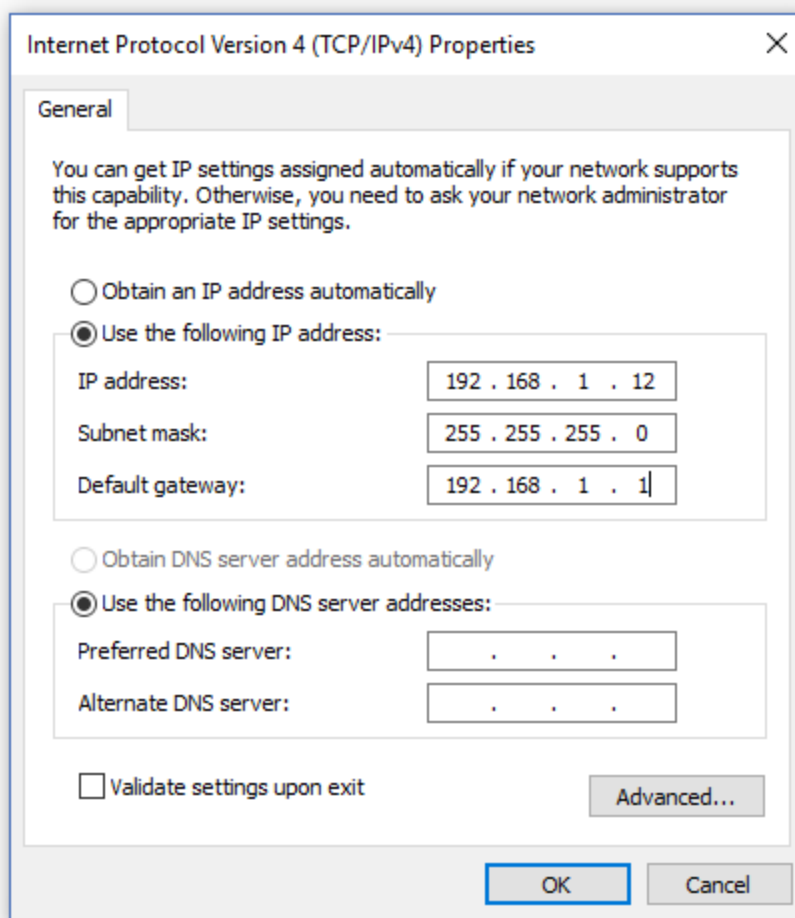
1. Go to Control Panel > Network and Sharing Center
2. Right-click your network adapter that connects to your Stack Controller and select Properties
3. Click Internet Protocol Version 4 (TCP/IPv4) and click Properties





**Figure 42. Network Adapter Properties on Windows.**

4. Update your network adapter TCP/IPv4 settings to the following:
  - Static IP Address: The IP address must have the same Network ID as the Stack Controller and a Host ID that isn't already in use.
    - Refer to the instruction above to determine a valid IP address for the PC.
  - Subnet Mask: 255.255.255.0
  - Default gateway: The Network ID must be the same as the Stack Controller and the Host ID should be 1.
    - For example, if the Stack Controller IP address is 192.168.1.21, the gateway address would be 192.168.1.1



**Figure 43. Network Adapter Settings on Windows**

5. Ensure your computer is on the same network as your Stack Controller.

- You can connect an Ethernet cable directly between it and the network adapter of your PC.

1. Open the Nuvation-Energy-Operator-Interface.html file in your web browser.

- This is usually done by double-clicking on the file. If your default browser is not a supported browser, you may need to specify the browser to use.
- For example, on Windows you may need to right click on the file, select Open with, and then choose Chrome or Firefox from the list.

## 5.3. Upgrading the Nuvation Energy BMS

The Operator Interface is packaged with the appropriate version of Nuvation Energy BMS firmware. The firmware for Nuvation Energy BMS can be upgraded using the Operator Interface.

To upgrade the firmware Nuvation Energy BMS must be in Service Lockout. Please see [Section 6.5, "Understanding the Service Lockout and Self Check"](#) for more details.

### 5.3.1. Upgrading the Firmware

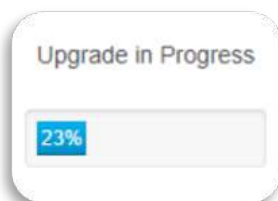
The Upgrade option will be available when Nuvation Energy BMS successfully enters Service Lockout. If a newer firmware version is available, proceed with the firmware upgrade.



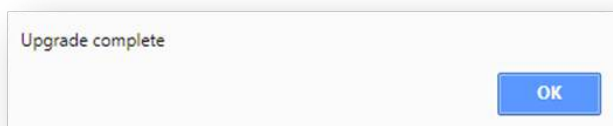
#### Backup the Config file

The firmware upgrade will erase the configuration file on your Nuvation Energy BMS. If this is not a new install and you have a configuration file loaded on your Nuvation Energy BMS, please remember to export and save it. You can import this file after the upgrade is complete. If upgrading to a new firmware version, you may need to edit the file for compatibility before uploading it.

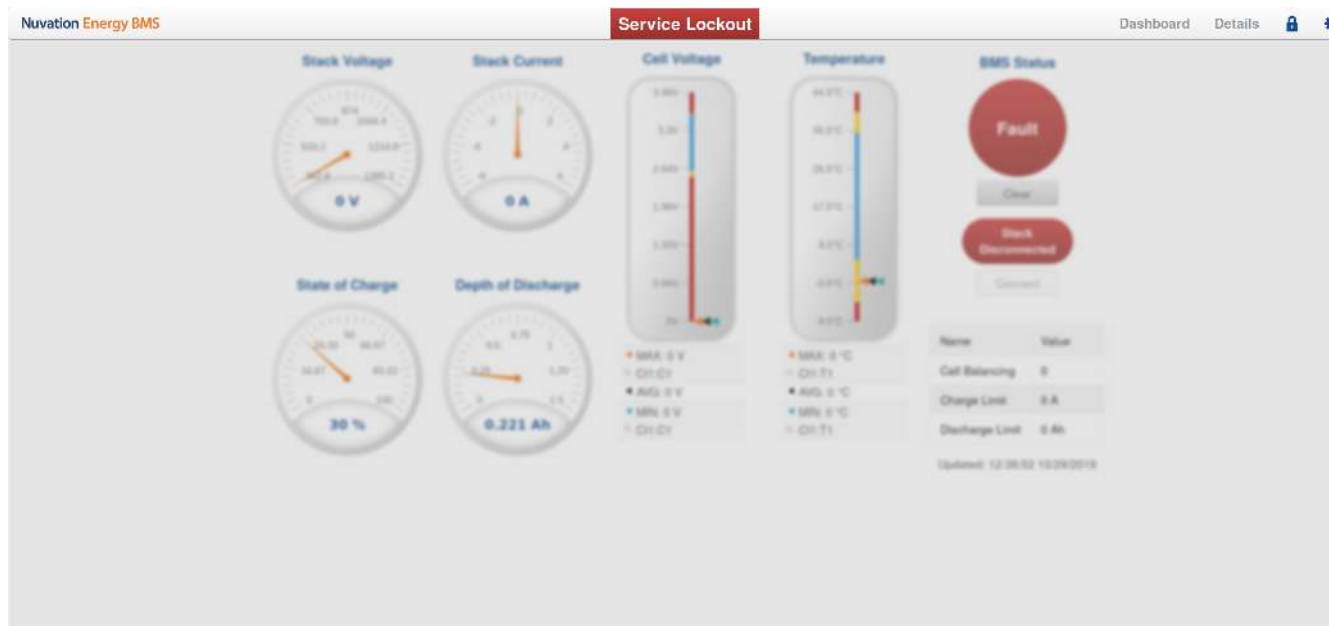
1. Click the Upgrade button to begin the upgrade. A progress bar allows you to monitor the upgrade process.



2. Wait until a dialog box appears with the upgrade result. It should indicate that the firmware upgrade is complete.



3. Once the upgrade is successful, you will be returned to the dashboard. Nuvation Energy BMS will remain in Service Lockout.



You will need to import a valid configuration file before attempting to exit Service Lockout.

## 5.3.2. Upgrade Troubleshooting

During the firmware upgrade, if a failure is reported, retry the upgrade.



If the failures persist, please submit a support ticket with as much detail as possible to [support@nuvationenergy.com](mailto:support@nuvationenergy.com).

## 5.4. Generate a Configuration File

Your Nuvation Energy BMS needs a valid configuration file to operate. The Operator Interface provides tools for importing and exporting configuration files to and from Nuvation Energy BMS as a way to set or retrieve the state of all configuration registers.

To generate a configuration file, visit the nCloud at: <https://ncloud.nuvationenergy.com>. nCloud is the Nuvation Energy online portal to remote battery management.

In the nCloud, configuration files can be generated and retrieved from the Configurations menu option by following these steps:

1. Click Create New Configuration
2. Follow the instructions in the Quick Start Wizard.
3. Download the .config configuration file
4. Store the configuration file on the computer running the Operator Interface.



**Figure 44. Configuration screen in the nCloud**



If your Nuvation Energy BMS is connected to a computer that doesn't have internet access, you can use a USB stick to transfer the configuration file between computers.

### 5.4.1. Fine-tuning the Configuration File (optional)

This is an optional step. If you would like to further adjust your Nuvation Energy BMS settings, to meet the requirements of your particular system, you may choose to edit the configuration file.

Refer to the [Section 7, "Configuration Settings"](#) for details on how to modify the configuration file for

your setup.

To edit the Configuration file, open the file in any text editor. Please backup the modified configuration file for future use. Nuvation Energy BMS doesn't store the file internally; it only stores the register values. The export feature will export a configuration file with an alphabetical listing of all registers and their set values.

## 5.5. Import the Configuration File

Once you have exported your configuration file to the computer connected to Nuvation Energy BMS, you can proceed to import it to Nuvation Energy BMS.

To import a configuration file Nuvation Energy BMS must be in Service Lockout. Please see [Section 6.5, "Understanding the Service Lockout and Self Check"](#) for more details.

1. Ensure you are in Service Lockout
2. Ensure Persist Configuration to BMS is checked



### Persist Configuration to BMS

Enabling this option tells Nuvation Energy BMS to persist the imported configuration file. If you are using known, good, configuration files you should check this box.

Disabling this option tells Nuvation Energy BMS to not persist the newly imported configuration file. On reboot, Nuvation Energy BMS will revert to the previous configuration file. This is useful when tweaking and testing configuration files. It allows you to recover from incorrect configuration settings by rebooting Nuvation Energy BMS

3. Click Import Configuration
4. Select the configuration file to use and click Open
5. A dialog indicating progress will pop-up



6. Wait until a dialog box appears with the configuration import result. It should indicate that the import was successful.



## 5.5.1. Configuration File Import Troubleshooting

During the configuration file import, if you receive a Register Write failure error, ensure you have valid entries in your configuration file. For details on the various registers and their intended use, please refer to [Section 7, "Configuration Settings"](#).

While exiting the service lockout, if a failure is reported, please see [Section 6.5, "Understanding the Service Lockout and Self Check"](#) for troubleshooting details.



If the failures persist, please submit a support ticket with as much detail as possible to [support@nuvationenergy.com](mailto:support@nuvationenergy.com).



## 5.6. Calibrate your Nuvation Energy BMS

Before you start regular operation of your Nuvation Energy BMS, it is recommended that you calibrate it for accurate usable capacity measurements. This calibration should be re-done if the Power Interface module or the current shunt is changed in the future.

1. Refer to [Section 7.9, "Measurement Calibration"](#) for steps to calibrate the stack current, voltage, and temperature measurements
2. Fully charge up the batteries to 100% State-of-Charge (as defined in the configuration file)
3. Discharge the batteries to 0% State-of-Charge (as defined in the configuration file)
4. Optionally, charge the batteries back up to 100% State-of-Charge

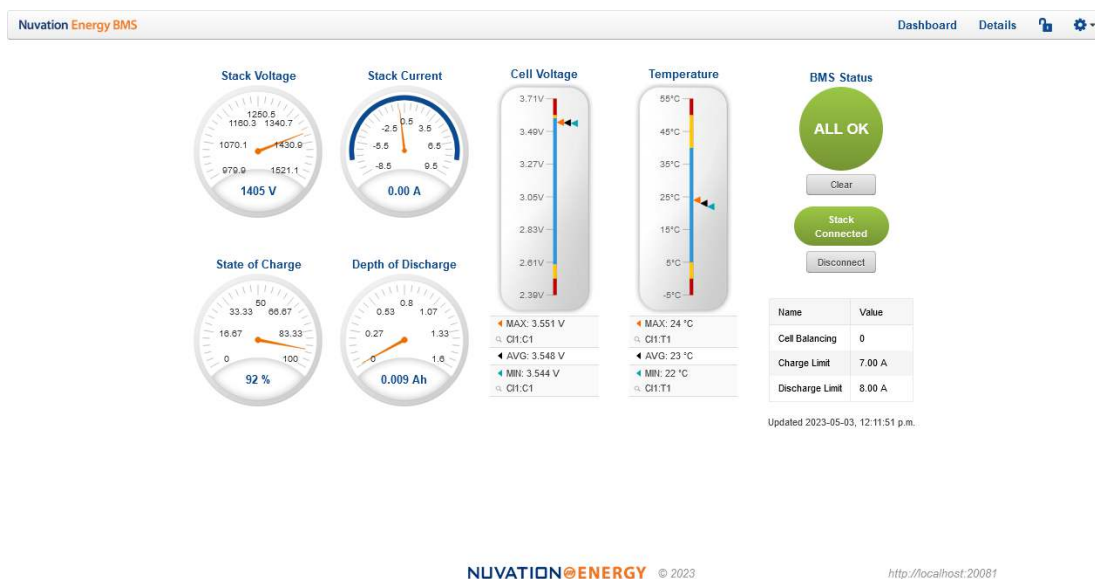


A power cycle during the State-of-Charge calibration steps will require the State-of-Charge calibration to be redone from the beginning.

## 6. Using the Operator Interface

### 6.1. The Dashboard Tab

The default tab of the Operator Interface is the Dashboard. The Dashboard contains a high-level overview on the state of the battery stack. This is the only page required for daily monitoring of the battery stack.



**Figure 45. Nuvation Energy BMS Operator Interface Dashboard screenshot**

#### 6.1.1. Warnings and Faults

Before going into the details of the gauges and information presented in the dashboard, it is important to understand what a fault and a warning Nuvation Energy BMS status means.



An ALL OK indicates that there are no faults or warning. This is the normal state for Nuvation Energy BMS.



A Warning indicates the state of the battery system has been detected outside of its normal operational range. The cause of the warning should be identified and a corrective action should be performed. For instance, if the warning is a thermistor temperature measurement has become too hot, the battery system should be cooled to bring the measurement back into the normal operational range.

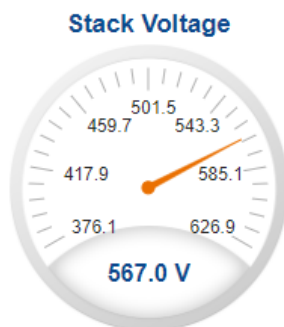


A Fault indicates the state of the battery system has been detected outside of its safe operational range. The cause of the fault must be identified and a corrective action must be performed. For instance, if the fault is a cell voltage measurement has become too low, the cell maintenance manual must be reviewed to identify what remedial actions are required.

A Fault is more severe than a Warning and the source of the fault must be discovered and resolved before attempting to clear and continue operating the battery system.

## 6.1.2. Stack Voltage

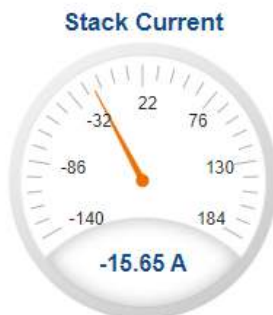
The stack voltage radial meter shows the total battery stack voltage.



## 6.1.3. Stack Current

The stack current radial gauge shows the battery stack current as well as the maximum charge current limit and the maximum discharge current limit. The acceptable current range is visualized on the gauge by the blue arc. An absence of the blue arc indicates the battery stack cannot be charged or discharged in its present condition.

A negative current value indicates the battery stack is charging. A positive current value indicates the battery stack is discharging.



## 6.1.4. State-of-Charge

The State-of-Charge radial gauge shows the battery stack's State-of-Charge. The battery stack is empty when the State-of-Charge value is 0% and full when the State-of-Charge value is 100%.

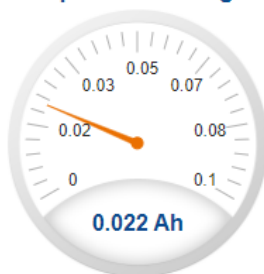
## State of Charge



## 6.1.5. Depth-of-Discharge

The Depth-of-Discharge radial gauge shows how much energy has been taken out of the battery stack. In an ideal Energy Storage System, defined as a system with no power losses, the amount of energy shown in this gauge needs to be added back into the battery stack to fill it back up to 100% State-of-Charge.

## Depth of Discharge



## 6.1.6. Cell Voltage

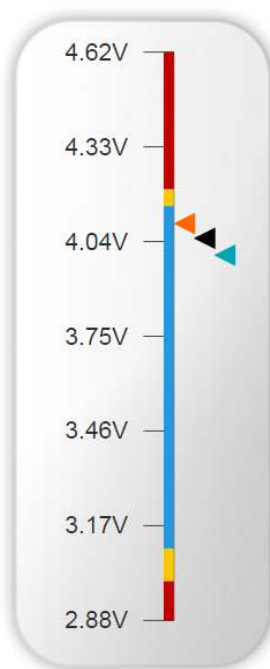
The cell voltage bar gauge shows the maximum, minimum, and average cell voltage measurements within the stack.

The high cell voltage and low cell voltage warning and fault threshold is visualized on the gauge with yellow and red segments. The blue segment depicts the acceptable cell voltage range.

If a triangle enters the yellow segment, a warning has occurred. If a triangle enters the red segment, a fault has occurred.

The maximum and minimum cell location in the stack and their voltage values are shown below the gauge, along with the average cell voltage value.

## Cell Voltage



◀ MAX: 4.095 V

🔍 CI1:C3

◀ AVG: 4.05 V

◀ MIN: 3.998 V

🔍 CI1:C1

## 6.1.7. Temperature

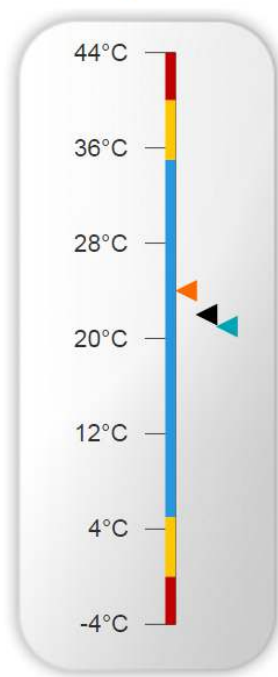
The temperature bar gauge shows the maximum, minimum, and average cell temperature measurements within the stack.

The high cell temperature and low cell temperature warning and fault threshold is visualized on the gauge with yellow and red segments. The blue segment depicts the acceptable cell temperature range.

If a triangle enters the yellow segment, a warning has occurred. If a triangle enters the red segment, a fault has occurred.

The maximum and minimum cell location in the stack and their temperature values are shown below the gauge, along with the average cell temperature value.

## Temperature



◀ MAX: 24 °C

🔍 [CI7:T3](#)

◀ AVG: 22 °C

◀ MIN: 21 °C

🔍 [CI10:T4](#)

## 6.1.8. Nuvation Energy BMS Status

Nuvation Energy BMS status information contains information on the overall safety status of the battery stacks, the battery stack connection state, number of cells balancing, maximum charge current limit, maximum discharge current, and the time and date of the last update of the Dashboard.

### 6.1.8.1. Operation Status

Nuvation Energy BMS operation state is shown in the big status circular indicator.



**Figure 46. Three possible Nuvation Energy BMS operation states**

The normal state is All OK and the color of the indicator will be green. The warning state is Warning and the color of the indicator will be orange. The fault state is Fault and the color of the indicator will be red.

Clicking on the indicator will jump to a comprehensive status list of warnings and faults active in the battery stack.

Clicking on the Clear button below the state will cancel any warnings and faults that are not self-clearing.

## 6.1.8.2. Connection State

The battery stack connection state is shown in the oval indicator.



**Figure 47. Three possible connection states**

Stack Disconnected in a red oval indicates the SSG contactors are open, and the battery stack is unavailable to be charged or discharged.

Stack Pre-charging in an orange oval indicates the battery stack has connected its pre-charge circuit and is attempting to equalize the battery stack voltage to the system DC bus voltage.

Stack Connected in a green oval indicates the battery stack is available to be charged or discharged.

Clicking the Connect button initiates the stack connection sequence of events. Nuvation Energy BMS must be in the All OK state for the Connect button to be available.

Clicking the Disconnect button will disconnect the battery stack from the system DC bus.

## 6.1.8.3. Information Table

The information table shows the number of cells that are having excess energy bled off to maintain a balanced battery stack.

Name	Value
Cell Balancing	0
Charge Limit	7 A
Discharge Limit	8 A

The Charge Limit shows the maximum charge current limit value. The Discharge Limit shows the maximum discharge current limit value.

The Charge Limit and Discharge Limit values are visualized on the Stack Current radial gauge as the limits of the blue arc.

#### 6.1.8.4. Last Update

The Updated time and date shows the last time the Operator Interface had successfully communicated with Nuvation Energy BMS and updated all items in the Dashboard with values from Nuvation Energy BMS. The time and date is based on the local computer/tablet; it does not come from Nuvation Energy BMS.

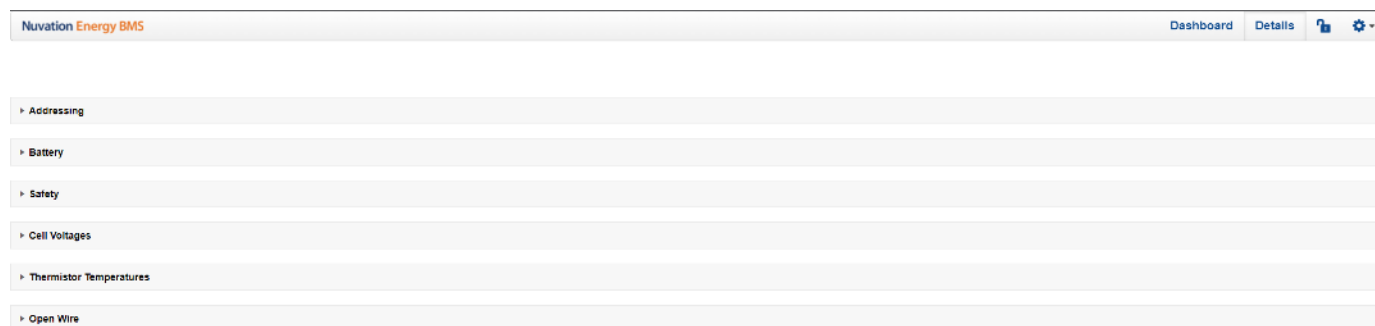
If the communication with a Nuvation Energy BMS is lost, a notification banner appears at the top of the display screen. The information shown on the Dashboard represents the last data received and is no longer recent. Refer to [Section 6.4.3, "Communication Loss"](#) for more details.



## 6.2. The Details Tab

The Details tab contains a much more detailed view into the status of Nuvation Energy BMS. The data values shown in this tab can be easily copied into a spreadsheet as a means of capturing the current state of Nuvation Energy BMS for manual data recording purposes.

The Details tab has multiple sub-sections called *accordions* that can be expanded to reveal more information. You can have multiple accordions expanded at the same time.



**Figure 48. Nuvation Energy BMS Operator Interface Details tab screenshot**

### 6.2.1. Addressing

The Addressing accordion presents the addressing information for your Nuvation Energy BMS. Refer to [Appendix C](#) for details on how to change your Nuvation Energy BMS IP address.

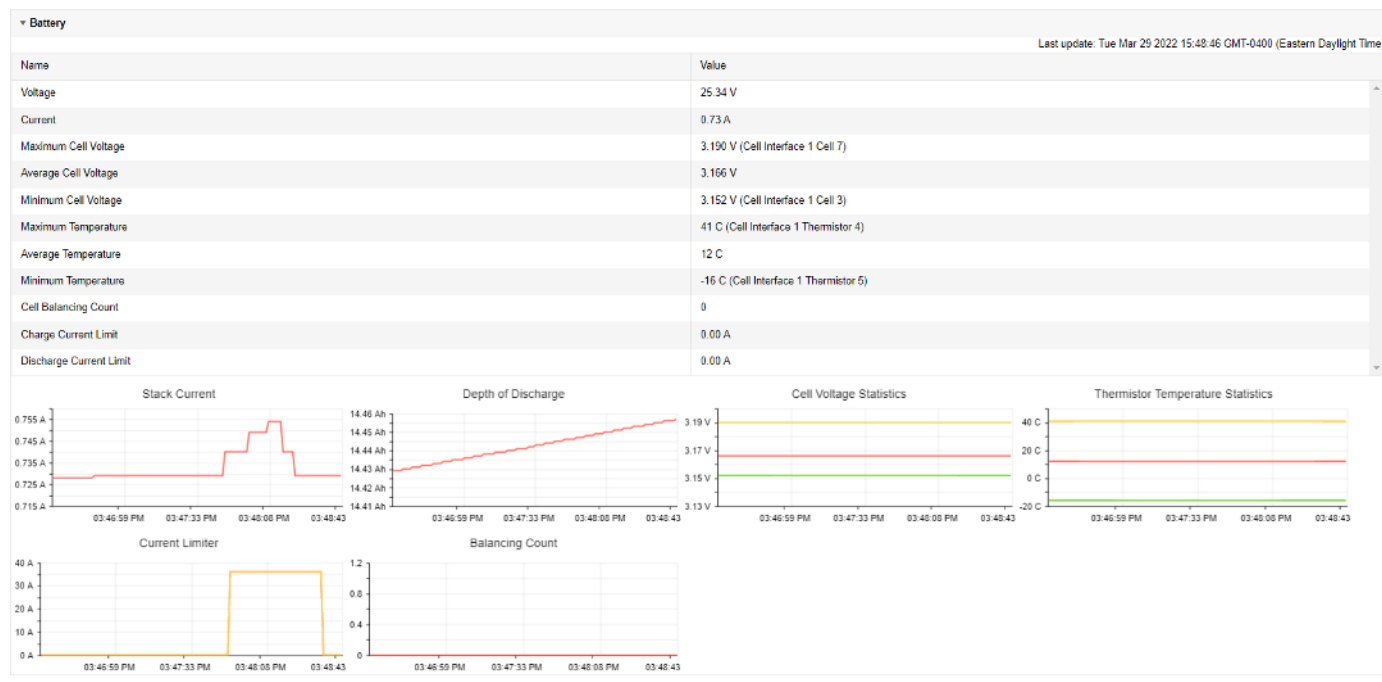
▼ Addressing		Last update: Fri Nov 29 2019 12:42:32 GMT-0500 (Eastern Standard Time)
Network	Address	
IP	192.168.1.21	
Modbus RTU	1	
CAN	10	

**Figure 49. Addressing accordion in Details Tab**

### 6.2.2. Battery

The Battery accordion contains values on the overall battery stack and the maximum charge current limit, the maximum discharge current limit, and the number of cells balancing in the battery stack. This information is identical to the values shown in the radial gauges, the bar gauges and the

Information Table on the Dashboard.



**Figure 50. Battery accordion in Details Tab**

## 6.2.3. Safety

The Safety accordion contains a comprehensive list of all possible Nuvation Energy BMS faults and warnings as well as the overall status of the battery stack. An active fault is shown as Tripped. An active warning or user trigger is shown as Triggered. A fault or warning that has not completed its self check is shown as Checking. In normal operation, all warnings and faults should be clear and the battery stack can be charged and discharged.

▼ Safety

Last update: Fri Mar 25 2022 17:25:44 GMT-0400 (Eastern Daylight Time)

Name	State	Trigger
Stack Under Voltage Fault	Clear	stack_fault_voltage_under
Stack Over Voltage Fault	Clear	stack_fault_voltage_over
Stack Low Voltage Fault	Clear	stack_fault_voltage_lo
Stack High Voltage Fault	Clear	stack_fault_voltage_hi
Cell Under Voltage Fault	Clear	stack_fault_cell_under
Cell Over Voltage Fault	Clear	stack_fault_cell_over
Cell Low Voltage Fault	Clear	stack_fault_cell_lo
Cell High Voltage Fault	Clear	stack_fault_cell_hi
Discharge Under Temperature Fault	Clear	stack_fault_discharge_therm_under
Discharge Over Temperature Fault	Clear	stack_fault_discharge_therm_over

**Figure 51. Safety accordion in Details Tab**

Clicking on the Clear Faults and Warnings button at the bottom of this accordion will clear any faults and warnings that are not self-clearing. The Clear button on the Dashboard can also be used.

Stack Under Voltage Lockout	Clear	stack_uvlo_clear_voltage
Cell Under Voltage Lockout	Clear	stack_uvlo_cell_voltage
Discharge High Current User	Clear	stack_trig_discharge_current_hi
Charge High Current User	Clear	stack_trig_charge_current_hi
Discharge Low Temperature User	Clear	stack_trig_discharge_therm_lo
Discharge High Temperature User	Clear	stack_trig_discharge_therm_hi
Charge Low Temperature User	Clear	stack_trig_charge_therm_lo
Charge High Temperature User	Clear	stack_trig_charge_therm_hi
Stack Low Voltage User	Clear	stack_trig_voltage_lo
Stack High Voltage User	Clear	stack_trig_voltage_hi
Cell Low Voltage User	Clear	stack_trig_cell_lo
Cell High Voltage User	Clear	stack_trig_cell_hi
Stack Controller Configuration Consistency Check Fault	Clear	sc_fault_config

Clear Faults and Warnings      Generate Report

**Figure 52. Bottom of Safety accordion in Details Tab**

Clicking on the Generate Report button at the bottom of this accordion will generate a safety report JSON file with a list of the Nuvation Energy BMS faults and warnings and their current state as well as the current firmware versions.

## 6.2.4. Cell Voltages

The Cell Voltages accordion lists voltage measurements for all Cells configured in the Configuration file. Cells that are not configured are displayed as a - (hyphen). Voltages in red indicate measurements which have triggered a Nuvation Energy BMS fault. Voltages that are highlighted in yellow are open wires. There is no differentiation between cells that are in the normal operating voltage range and cells that have triggered a Nuvation Energy BMS warning. There is also no indication of which particular cells are currently being balanced by the BMS.

Cell Voltages																
Last update: Fri Mar 25 2022 16:21:03 GMT-0400 (Eastern Daylight Time)																
Filter:	Off	Above	Below													
	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8	Cell 9	Cell 10	Cell 11	Cell 12	Cell 13	Cell 14	Cell 15	Cell 16
Cl 1	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-
Cl 2	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-
Cl 3	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-
Cl 4	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-
Cl 5	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-
Cl 6	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-
Cl 7	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-
Cl 8	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-

**Figure 53. Cell Voltages accordion in Details Tab**

Cell Voltages

Filter:  Above Below  
Legend:

Last update: Wed Nov 20 05:05:14 GMT-0500 (Eastern Standard Time)

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8	Cell 9	Cell 10	Cell 11	Cell 12	Cell 13	Cell 14	Cell 15	Cell 16
CI 1	3.335 V	3.333 V	3.332 V	3.333 V	3.332 V	3.332 V	3.333 V	3.333 V	3.334 V	3.331 V	3.331 V	3.333 V	-	-	-	-
CI 2	3.334 V	3.333 V	3.333 V	3.334 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.334 V	3.334 V	-	-	-	-
CI 3	3.331 V	3.333 V	3.334 V	3.333 V	3.334 V	3.333 V	3.333 V	3.333 V	3.334 V	3.334 V	3.331 V	3.334 V	-	-	-	-
CI 4	3.333 V	<b>3.335 V</b>	3.334 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.332 V	3.334 V	-	-	-	-
CI 5	3.334 V	3.333 V	3.334 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-
CI 6	3.333 V	3.333 V	3.332 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.334 V	3.333 V	3.333 V	3.333 V	-	-	-	-
CI 7	3.334 V	3.333 V	3.333 V	<b>3.335 V</b>	3.334 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-
CI 8	3.333 V	3.334 V	3.333 V	3.334 V	3.333 V	3.333 V	3.333 V	3.334 V	3.333 V	3.333 V	3.333 V	3.333 V	-	-	-	-

Figure 54. Cell Voltages with open wires according in Details Tab

### 6.2.4.1. Filtering

You can filter the display to highlight cells with voltages above or below a value you specify. The values that match the criteria will be bolded and all other values will be faded.

Filter:  Above Below

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8
CI 1	3.333 V	3.333 V	3.332 V	<b>3.335 V</b>	3.333 V	3.333 V	3.333 V	3.333 V

### 6.2.5. Thermistor Temperatures

The Thermistor Temperatures accordion lists temperature measurements for all Thermistors configured in the Configuration file. Thermistors that are not configured are displayed as a dash. Temperatures in red indicate measurements which have triggered a Nuvation Energy BMS fault. There is no differentiation between thermistors that are in the normal operating temperature range and thermistors that have triggered a Nuvation Energy BMS warning.

Thermistor Temperatures

Filter:  Above Below

Last update: Fri Nov 29 12:50:06 GMT-0500 (Eastern Standard Time)

	Index 1	Index 2	Index 3	Index 4	Index 5	Index 6	Index 7	Index 8
CI 1	27 °C	27 °C	27 °C	26 °C	27 °C	27 °C	-	-
CI 2	27 °C	27 °C	28 °C	27 °C	28 °C	27 °C	-	-
CI 3	26 °C	26 °C	27 °C	27 °C	27 °C	27 °C	-	-
CI 4	27 °C	27 °C	27 °C	27 °C	27 °C	27 °C	-	-
CI 5	28 °C	27 °C	27 °C	27 °C	27 °C	27 °C	-	-
CI 6	27 °C	27 °C	27 °C	28 °C	27 °C	27 °C	-	-
CI 7	26 °C	26 °C	26 °C	27 °C	27 °C	27 °C	-	-
CI 8	27 °C	27 °C	27 °C	28 °C	27 °C	27 °C	-	-

Figure 55. Thermistor accordion in Details Tab

### 6.2.5.1. Filtering

You can filter the display to highlight cells with temperatures above or below a value you specify. The values that match the criteria will be bolded and all other values will be faded.

Filter:  Above Below

	Index 1	Index 2	Index 3	Index 4	Index 5	Index 6	Index 7	Index 8
CI 1	27 °C	27 °C	27 °C	<b>26 °C</b>	27 °C	27 °C	-	-

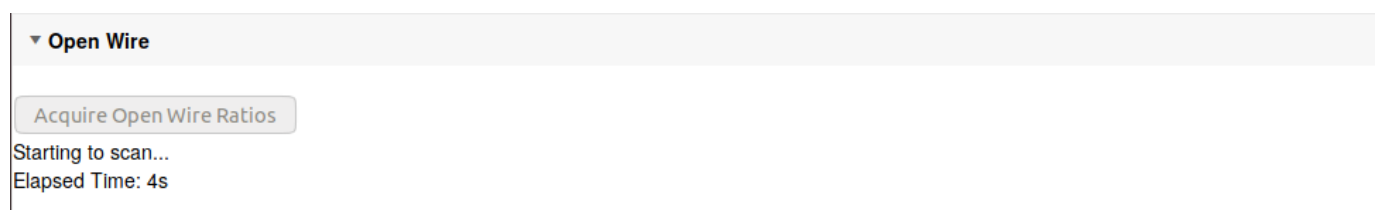
### 6.2.6. Open Wire

The Open Wire accordion lists open wire diagnostics for all Cells configured in the Configuration file. Cells that are not configured are displayed as a - (hyphen). Diagnostic data that is highlighted in yellow indicates an open wire.



**Figure 56. Open Wire accordion in Details Tab**

To trigger diagnostics on all cells, click the "Acquire Open Wire Ratios" button. Once clicked, the open wire scanning process begins.



**Figure 57. Open Wire acquiring accordion in Details Tab**

Once the open wire scanning has completed and all diagnostics data is collected it is displayed in a tabular format with open wires highlighted in yellow.

▼ Open Wire															
<div>Acquire Open Wire Ratios</div> <div>Buttons getting open wire ratios</div> <div>Filter: <b>Cell</b> <b>Device</b> <b>Details</b></div> <div>Legend: <b>0.00 - 0.99</b></div>															
	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8	Cell 9	Cell 10	Cell 11	Cell 12	Cell 13	Cell 14	Cell 15
Cell 1	0.920	0.921	0.922	0.922	0.919	0.919	0.921	0.920	0.920	0.921	0.916	0.920			
Cell 2	0.919	0.922	0.924	0.924	0.924	0.925	0.925	0.925	0.924	0.925	0.923	0.921			
Cell 3	0.920	0.920	0.922	0.924	0.924	0.923	0.923	0.924	0.925	0.924	0.922	0.920			
Cell 4	0.922	0.923	0.921	0.924	0.924	0.923	0.923	0.923	0.922	0.923	0.924	0.923			
Cell 5	0.920	0.922	0.921	0.921	0.922	0.922	0.920	0.920	0.922	0.921	0.920	0.920			
Cell 6	0.905	0.905	0.900	0.900	0.909	0.909	0.909	0.903	0.908	0.908	0.908	0.907			
Cell 7	0.910	0.912	0.912	0.912	0.910	0.910	0.912	0.912	0.912	0.912	0.912	0.910			
Cell 8	0.910	0.911	0.913	0.914	0.915	0.915	0.914	0.901	0.902	0.918	0.914	0.914			

**Figure 58. Open Wire diagnostics accordion in Details Tab**

The diagnostic values displayed are ratios of Voltage. Values very close to one ( > 0.97) indicate a short condition. Values approaching zero (0.0 to 0.4) indicate an open wire connection.

Depending on whether you are using Cell Interface - 12 channel or Cell Interface - 16 channel modules or Cell Interface - 12 V 4 channel modules, they will display different diagnostics information.

## 6.2.6.1. Filtering

Use the filter to highlight ratios of Voltage that are above or below the specified value. The values that match the criteria will be bolded and all other values will be faded.

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Cell 7	Cell 8
Cl 1	0.900	<b>0.920</b>	0.600	0.900	<b>0.920</b>	0.600	0.600	0.800

### 6.2.7. SoC Calibration

The State of Charge Calibration accordion allows for calibration of SoC and reset of the measured capacity estimate. This accordion is only visible if the OCV is enabled (`stack_soc.ocv_enabled`). Both of these operations are only enabled when the battery is operational (not in service lockout) but disconnected (`stack_control.actual_state == 0`).

▼ SOC Calibration

Last update: Fri Mar 25 2022 16:27:06 GMT-0400 (Eastern Daylight Time)

Battery Charge	Battery Capacity
State of Charge      67 %	Measured Capacity      0.000 Ah
Depth of Discharge      0.150 Ah	Nameplate Capacity      1.500 Ah
<div style="margin-bottom: 10px;"><button type="button" value="Calibrate"></button></div> <p>SOC is calibrated from cell voltages using the open-circuit voltage curve. Battery contactors must be disconnected to calibrate.</p>	<div style="margin-bottom: 10px;"><button type="button" value="Reset"></button></div> <p>Capacity is reset to stack nameplate rating while accounting for cell SOC imbalance. Battery contactors must be disconnected to reset.</p>

### Figure 59. SoC Calibration accordion in Details Tab

## 6.3. The Menu Options

The menu to the right of the Operator Interface provides access to tools and advanced options. Some of these options may be locked to prevent accidental changes.

The following options are available:

- Registers
- Service
- Connection
- Unlock
- About

### 6.3.1. Registers

This menu option navigates to the Register Browser. This screen allows you to read and write to registers.

When the Operator Interface is locked, the Register Browser is in a view-only mode, restricting the editing of registers.

Component Name	Description	Base Address	Instance Count per Stack
sc_clock	Clocking Component	0x29000	1

**Figure 60. Register Browser**

### 6.3.2. Service

This screen allows you to enter and exit Service Lockout in-order to perform an upgrade or configuration file import. For more details refer to [Section 6.5, "Understanding the Service Lockout and Self Check"](#).

#### 6.3.2.1. Importing a Configuration File



This is a restricted option requiring the Operator Interface to be unlocked to access this option.

This menu option allows you to import a configuration file.

If you have a configuration file ready to import, refer to [Section 5.5, "Import the Configuration File"](#) for instructions.

If you need to generate a configuration file, refer to [Section 5.4, "Generate a Configuration File"](#) for instructions.

## 6.3.2.2. Exporting a Configuration File



Nuvation Energy BMS does not preserve the originally imported configuration file with comments and formatting. The export feature will export a configuration file with an alphabetical listing of all registers and their set values.

To export your configuration file:

1. From the menu, select Service
2. Click Export Configuration

## 6.3.2.3. Upgrade



This is restricted option requiring the Operator Interface to be unlocked to access this option.

Refer to [Section 5.3, "Upgrading the Nuvation Energy BMS"](#) for instructions on performing a firmware upgrade.



The firmware upgrade will erase the configuration file on your Nuvation Energy BMS. Please remember to export and save your current configuration file to save your current configuration file for future reference.

## 6.3.2.4. Updating a Label

A label is the string displayed on the top banner of the Operator Interface. An owner can configure a name for each stack using the label. When the Operator Interface connects to a stack the operator knows immediately what stack they are connected to.



To update a label:

1. From the menu, select Service
2. Click Edit under the label section



**Label**

The label is used to identify which BMS you are connected to.

Edit
Apply

3. Enter a suitable name for the stack

4. Click Apply



The Operator Interface automatically saves the label in persistent storage when applied.



A label does not handle spaces. Use hyphens or underscores to delineate words.



## 6.3.2.5. Editing an IP Address

To edit an IP address:

1. From the menu, select Service
2. Click Edit under the ethernet section

**Ethernet**

The local IP network configuration used by this BMS.

IP Address  
0.0.0.0

Netmask  
0.0.0.0

Gateway  
0.0.0.0

Edit
Apply

3. Enter the IP address of your choice

4. Click Apply



The updated IP is saved and in immediate use after the Apply action. Any prior connections via Modbus TCP would have to be reconnected using the updated IP.



Applying changes will disconnect the battery and restart the BMS.

### 6.3.3. Connection

The connection page allows you to change the IP address and timeout of the Nuvation Energy BMS that you are trying to access.

If you are connecting to a Nuvation Energy BMS that does not use the default IP address (192.168.1.21), you will need to change the connection settings in the Operator Interface to match the BMS.



In scenarios where there are multiple BMS systems, it is possible to direct multiple instances of the Operator Interface at the different BMS IP addresses.

### 6.3.4. Locking and Unlocking

The Operator Interface can be locked to prevent accidental or unintentional changes that could have serious impacts on the system. To further secure the system, a password can be set to ensure only authorized individuals can access these critical sections of the Operator Interface

When the Operator Interface is unlocked, the lock indicator will be replaced with an unlocked indicator.



**Figure 61. Operator Interface Locked indicator**



**Figure 62. Operator Interface Unlocked indicator**

#### 6.3.4.1. Lock the Operator Interface

To lock the Operator Interface, simply click the Unlocked indicator or bring up the settings menu and select Lock.



To prevent accidental changes to your Nuvation Energy BMS, always lock the Operator Interface after making your changes.

It is possible to require a password to unlock the Operator Interface.

## 6.3.4.2. Unlock the Operator Interface

To Unlock the Operator Interface, simply click the Locked indicator or bring up the settings menu and select Unlock. If a password has been set, the correct password will need to be entered to complete the unlocking process.

## 6.3.5. About

This screen displays version details for the underlying software for your Nuvation Energy BMS. The name and number of the current software release are displayed at the top of the About screen.

In the screenshot below, the release name is at the top in large blue letters. The version number following the release name has a format of yy.mm with yy representing the year and mm representing the month within that year that this Nuvation Energy BMS package was released.

The version numbers below the release name are the version numbers of the individual software packages running on your Nuvation Energy BMS.

When contacting support, please include the details in your About screen.

# About This BMS

Software Version	
OI:	0.57.0
SC:	4.109.1:01f91781 (4.109.1) (CRC 60404)
SC BL:	4.109.1 (CRC 19929)
SC Revision ID:	A2
PI:	4.109.1 (4.109.1) (CRC 16898)
PI BL:	4.109.1 (CRC 57871)
PI Revision ID:	Y
Factory Lockdown	
SC Factory Persist:	CRC 58422
PI Factory Persist:	CRC 5938

**Figure 63. Sample About Screen**



Not all information is displayed unless the BMS is factory locked. Additional information for identifying the configuration and images is provided to ensure the correct version of firmware and configuration is being used as specified at the

factory.

## 6.4. The Status Banner


The banner at the top of the screen is used to indicate high level changes in system status. During typical operation nothing is displayed and this indicates the Operator Interface is communicating with the BMS and the BMS is fully operational with no major diagnostics problems.

### 6.4.1. Service Lockout Indication

When the system is either entering or is in service lockout, a banner is displayed indicating this to the user.

A red rectangular banner with the text "Service Lockout" in white, bold, sans-serif font.


**Figure 64. Operator Interface Service Lockout indicator**

A red rectangular banner with the text "Entering Lockout" in white, bold, sans-serif font.

**Figure 65. Operator Interface Entering Service Lockout indicator**

### 6.4.2. Self Check Indication

When the system is either exiting service lockout or has just booted, the system will execute a Self Check to verify a safe operation. A banner is displayed indicating the Self Check execution to the user.

A red rectangular banner with a white circle icon containing a checkmark, followed by the text "Self-Check" in white, bold, sans-serif font.

**Figure 66. Operator Interface Self Check indicator**

### 6.4.3. Communication Loss

When the Operator Interface can no longer communicate with the Battery Management System, a communication lost banner is displayed at the top of the display. It will provide the amount of time since disconnected and updates each second.

A red rectangular banner with the text "Communication to BMS lost: Data last updated 4 seconds ago" in white, bold, sans-serif font.

**Figure 67. Operator Interface Communication lost indicator**

## 6.5. Understanding the Service Lockout and Self Check

Service Lockout allows you to put your Nuvation Energy BMS into a Lockout state while you perform maintenance on your energy storage installation such as loading new configurations, or upgrading the firmware.

If you are making physical wiring changes, you should enter Service Lockout before powering down Nuvation Energy BMS.

When in Service Lockout, Nuvation Energy BMS will set all faults and warnings as tripped.

A Self Check is a self-test that is executed when:

1. Exiting Service Lockout
2. When booting the Battery Management System

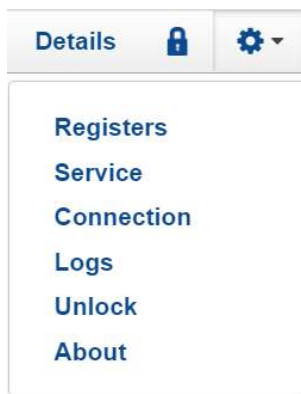
A Self Check is a verification to make sure that the configured Battery Management System is ready for operation. A Battery Management System remains in Self Check until the issues preventing a successful self-check are corrected. Refer to [Section 6.5.3, "Self Check"](#) for further details about the self check.



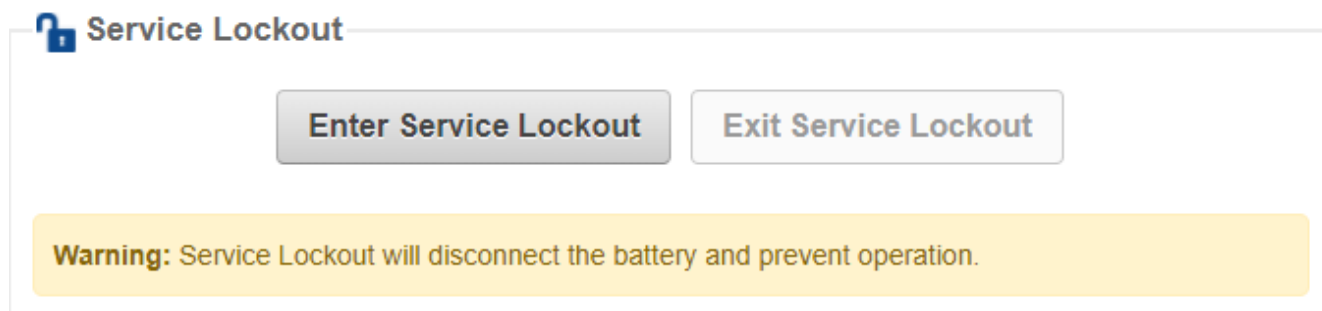
The status for the Service Lockout and Self Check are displayed in the status banner of the Operator Interface as indicated in [Section 6.4, "The Status Banner"](#)

### 6.5.1. Entering Service Lockout

1. Unlock the Operator Interface by clicking the Lock indicator and enter the password if prompted.
2. From the menu, select Service to bring up the Service page



3. Click Enter Service Lockout to enter Service Lockout



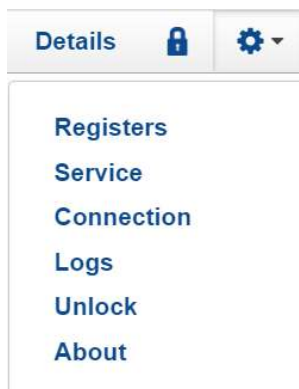
Entering Service Lockout will open all contactors and GPOs



When in Service Lockout, you will not be able to access the Operator Interface dashboard.

## 6.5.2. Exiting Service Lockout

1. Unlock the Operator Interface by clicking the Locked indicator and enter the password if prompted.
2. From the menu, select Service to bring up the Service page



3. Exit the Service Lockout by clicking Exit Service Lockout



Exiting Service Lockout may close contactors

4. A Self Check will be initiated and its banner will be displayed on the Operator Interface. Refer to [Section 6.4.2, "Self Check Indication"](#) for banner details.

## 6.5.3. Self Check

The Self Check verifies that all configured data inputs for the faults and warnings of the Battery Management System are being received. The Battery Management System will remain in Self Check until all these data inputs are received. The Self Check is executed continuously unless the Battery Management System is returned back to Service Lockout. If the issue(s) that were preventing a

successful Self Check are corrected, the Battery Management System will enter operation. You will be able to access the dashboard controls once Nuvation Energy BMS has successfully completed the Self Check.

## 6.5.4. Self Check Troubleshooting

If Nuvation Energy BMS fails its self check, you will receive an error notification. The reason for the failure is that some form of input to the Battery Management System was not received. This means that the Battery Management System was expecting some measurement (e.g. cell voltage, current) but it never was received by the software. Because of this lack of data, the Battery Management System should not be operated and the issue(s) needs to be resolved. There are two main causes for a lack of self-check failures:

- Misconfiguration: The current configuration does not match the Battery Management System deployment.
- Incomplete setup: There is an issue in the setup of the Battery Management System (e.g. cables are missing or faulty)

You can identify the parts of the Battery Management System that have not completed their self-check by referring to the Details > Safety accordion section. The faults/warnings that are currently failing self-check are indicated by the red text "Checking" on the right side of the display.

Refer to [Section 10.1, "Faults and Self Check Failures"](#) for details on how to address the self-check failures.

If you just imported a configuration file, re-import a known-good configuration file. You can download a previously generated configuration file from the nCloud (<https://ncloud.nuvationenergy.com>).



If the failures persist, please submit a support ticket with as much detail as possible to [support@nuvationenergy.com](mailto:support@nuvationenergy.com).



## 7. Configuration Settings

The following sections break down a complete system configuration into the major areas of responsibility.

### 7.1. Background and Terminology

Terminology and technical concepts critical to the operation and configuration of Nuvation Energy BMS are presented in this section.

#### 7.1.1. Register Data Model

##### 7.1.1.1. Registers and Components

Understanding the Register Data Model is key to understanding how to configure Nuvation Energy BMS.

Nuvation Energy BMS implements all data storage and processing using two important software building blocks.

##### **Register**

A register is the fundamental unit of data storage within the system. Each register has a unique name and associated type that defines how the value is interpreted. Registers range in size from as small as one byte up to as large as eight bytes.

##### **Component**

A component combines a set of related registers with processing rules that operate on those registers to implement a particular BMS function for the system. A given component may have many instances throughout the system. In this case, its associated registers will have the same number of instances.

Complex behavior within the system is achieved by connecting multiple components together, either through configuration or through hard-wired connections in the firmware itself.

Configuration for a system is completely determined by the state of all configuration registers present within the system. Configuration registers are persisted in non-volatile memory and are loaded automatically upon reset.

External protocols are implemented by mapping (and in some cases aggregating) the appropriate BMS registers to CAN bus messages or Modbus registers.

##### 7.1.1.2. Index versus Location

Internally to Nuvation Energy BMS firmware, all register indexing is zero-based. That is, if multiple instances of the same register are present, the first instance is always indexed at zero. This convention is reflected in all register expressions and configuration files.

Operator-facing tools such as Nuvation Energy BMS Operator Interface or the MESA Modbus models

use one-based location identifiers to refer to physical, countable entities.

For example, the location of the first cell within a stack is defined as Cell Interface 1, Cell 1—i.e. it is the first cell in the first Cell Interface. The index of this first cell is defined as zero within the firmware.

### 7.1.1.3. Register Expressions

Registers are accessed by name in Nuvation Energy BMS tools and configuration. Each register also has a unique address that is used internally within the BMS.

#### 7.1.1.3.1. Single Register Instance

This expression is used when assigning to or reading from a single register in the system and is of the form:

```
component_name.register_name
```

where:

- **component\_name** is the name of the component within the system
- **register\_name** is the name of the register within the component

#### 7.1.1.3.2. Range of Register Instances

These expressions build on the single register references by adding an additional range expression in square brackets:

```
component_name[range_expression].register_name
```

The **range\_expression** may take any of the following forms:

- **index** - A single instance of `component_name.register_name` at index. Note that `cell[0].voltage` is equivalent to `cell.voltage`.
- **start\_index:end\_index** - All instances from `start_index` through `end_index`. The expression `cell[0:3].voltage` expands into:

```
cell[0].voltage
cell[1].voltage
cell[2].voltage
cell[3].voltage
```

- **start\_index:end\_index:block\_length** - All instances from `start_index` through `end_index` within a repeating block of `block_length` across all instances of the register. For example if there were `N` instances of the register `cell.voltage`, the expression `cell[0:3:16].voltage` expands into:

```
cell[0].voltage
cell[1].voltage
cell[2].voltage
cell[3].voltage
cell[16].voltage
cell[17].voltage
cell[18].voltage
cell[19].voltage
cell[32].voltage
cell[33].voltage
cell[34].voltage
cell[35].voltage
...
cell[N-16].voltage
cell[N-15].voltage
cell[N-14].voltage
cell[N-13].voltage
```

- **start\_index:end\_index:block\_length:block\_count** - All instances from start\_index through end\_index within a repeating block of block\_length repeated block\_count times. The expression `cell[0:3:16:2].voltage` expands into:

```
cell[0].voltage
cell[1].voltage
cell[2].voltage
cell[3].voltage
cell[16].voltage
cell[17].voltage
cell[18].voltage
cell[19].voltage
```

In this case, only the first 2 blocks of 16 instances are included, rather than all blocks of 16 instances.

### 7.1.1.3.3. All Register Instances

A compact syntax can be used to expand to all instances of a given register within the system. The expression:

```
component_name[*].register_name
```

expands to all instances of **component\_name.register\_name** within the system. For example, the expression `cell[*].voltage` expands into:

```
cell[0].voltage
cell[1].voltage
cell[2].voltage
```

```
cell[3].voltage
cell[4].voltage
cell[5].voltage
cell[6].voltage
cell[7].voltage
cell[8].voltage
cell[9].voltage
cell[10].voltage
cell[11].voltage
cell[12].voltage
cell[13].voltage
cell[14].voltage
cell[15].voltage
cell[16].voltage
.
.
.
cell[N-3].voltage
cell[N-2].voltage
cell[N-1].voltage
```

where N is the total number of instances of the register `cell.voltage` within the system.

#### 7.1.1.3.4. Register Address

In some cases, it is necessary to use the address of a register as a configuration value for another register in the system. This is required when assigning input and output pins to functions within the BMS, for example.

The expression:

```
@component_name.register_name
```

expands to the address of the register in the system. The single-instance range expression may be used for register addresses. For example:

```
@component_name[index].register_name
```

expands to the address of **component\_name.register\_name** at index in the system.

#### 7.1.2. Units

A standard set of types/units has been adopted for use within Nuvation Energy BMS for the measurements and configuration settings. Unless otherwise noted, the units used within the firmware should be assumed as defined below.

**Table 3. Standard Types and Units**

Type	Units	Variable Type	Application
Address	none	uint32	BMS register address
Boolean	false=0, true=1	uint8	Logic value
Charge	mAh	int32	Depth of discharge and throughput
Count	number	uint32	A positive count of some event
Crc16	none	uint16	Used to verify persistent data
Current	mA	int32	Stack and pack currents
Energy	Wh	uint32	Energy capacity of a stack
Int64	value	int64	Signed 64 bit integer
ImpedanceMili	mΩ	uint32	Resistance of a stack
IpAddress	IP	uint32	IP4 address for a stack
LogLevel	none	uint32	Log level applied the BMS firmware
MicroOhms	μΩ	uint32	Resistance of a cell
Microseconds	μs	uint64	Time measurement or period
Percentage	%	uint8	Percentage of a full scale value (e.g. SoC)
PowerW	W	int32	Rated power of a stack
SoftwareId	enum	uint32	Enumeration for type of Cell Interface
Temperature	°C	int16	Thermistor temperatures
String	none	char[8]	Storage for 8 characters
UInt16	value	uint16	Unsigned 16 bit integer
UInt64	value	uint64	Unsigned 64 bit integer
Voltage	mV	int32	Cell and stack voltages

Every register within the firmware has an associated type that defines the expected units for that register.



A negative value for *Current* represents a charge direction while a positive value represents a discharge direction with respect to the cell/stack. Some *Current* registers represent the magnitude of the current and are never negative in value. These register exceptions include the following:

- `stack_current_limit.min_charge_current`
- `stack_current_limit.max_charge_current`
- `stack_current_limit.max_discharge_current`
- `stack_charge_status.hold_current`
- `stack_control.precharge_max_current`
- `stack_control.precharge_bias_current`
- `stack_fault_precharge_over_current.thresh`

## 7.1.3. Storage Types

Each register in Nuvation Energy BMS can be one of five storage types.

### **Volatile**

- Data that is not persisted across a system reset

### **Configuration**

- Data that is persisted on the [service page](#) via loading a configuration file.

### **Persistent**

- Data that is persisted and regularly saved

### **Factory**

- Data that is persisted when requested and is typically set at the factory

### **Manufacturing**

- Data that is persisted and saved at manufacturing time

These types of storage can be read directly from the Operator Interface *Registers* menu option as discussed in [Section 6.3.1, "Registers"](#).

## 7.1.4. Configuration File

Configuration is stored externally to Nuvation Energy BMS in a plain-text file. This file defines the state of configuration registers as required for a particular system.

To generate a configuration file for a stack, visit the nCloud at: <https://ncloud.nuvationenergy.com>. nCloud is the Nuvation Energy online portal.

The Operator Interface provides tools for importing and exporting configuration files to and from Nuvation Energy BMS as a way to set or retrieve the state of all configuration registers.

The configuration file format is plain ASCII text with the following syntax:

- Any lines starting with a leading # are treated as comments.
- Each non-comment line is treated as a register assignment statement.

A register assignment takes on the form `register_expression = value` where:

- `register_expression` is one of the valid [Register Expressions](#) previously defined.
- `value` is either a numerical constant, quoted string, IP address, or a valid [Register Address](#).

Any standard text editor can be used to edit configuration files (e.g. Notepad++, etc.).



A configuration file cannot be used to configure the IP settings of a stack. Stack IP configuration is detailed in the [Appendix C](#).



When a Battery Management System has been configured for [Factory Lockdown](#)

there are a set of components that will have all their register protected from most write operations. Attempting to perform a write operation via a register addressing (for example using a [Digital Inputs](#)) will silently fail. Please refer to [Section 7.2.4, "Restricted Register Write Operations"](#) for further details.

## 7.2. Factory Lockdown

*Factory Lockdown* is a mechanism for users to lock registers with *Factory* storage type as read-only. This is typically used to protect safety related configurations for safety certification purposes.

### 7.2.1. Configuring Factory Registers

Factory registers can be configured in the same way normal configuration registers can be set. They can be set through the Operator Interface *Registers* menu option ([Section 6.3.1, "Registers"](#)) or applied by uploading a configuration file. Factory registers can only be configured when the system is not in *Factory Lockdown*.

### 7.2.2. Enabling Factory Lockdown

*Factory Lockdown* can be enabled by writing to specific registers from the Operator Interface. The user must follow the process below for the Nuvation Energy BMS to be considered factory-locked.

1. Ensure all factory registers are configured to their desired value
2. Write 1 to `sc_factory_lockdown.requested_state`
3. Write 1 to `sc_factory_persist.save`
4. Wait until `sc_factory_persist.save` is equal to 0
5. Verify `sc_factory_lockdown.actual_state` is equal to 1
6. Write 1 to `pi_factory_lockdown.requested_state`
7. Write 1 to `pi_factory_persist.save`
8. Wait until `pi_factory_persist.save` is equal to 0
9. Verify `pi_factory_lockdown.actual_state` is equal to 1

### 7.2.3. Clearing Factory Lockdown

A system can only exit *Factory Lockdown* through a firmware upgrade or a factory reset.

For performing a factory reset, refer to [Appendix D](#).

### 7.2.4. Restricted Register Write Operations

When *Factory Lockdown* has been enabled, all register writes are restricted through external interfaces (e.g. Modbus-TCP, HTTP) with some exceptions. A 'whitelist' of registers which can be written to when in *Factory Lockdown* and are excluded from write protection are listed in [Table 4](#).

["Factory Lockdown Whitelist".](#)

**Table 4. Factory Lockdown Whitelist**

Register Name	Register Address
sc_op_state.requested_state	279776
sc_boot.requested_state	229376
stack_trigger_summary.clear_faults	131081
stack_trigger_summary.clear_warnings	131082
stack_factory_trigger_summary.clear_faults	221449
stack_factory_trigger_summary.clear_warnings	221450
sc_linkbus.cell_open_wire_manual_scan	69650
sc_linkbus.diagnostics	69653
stack_control.requested_state	200705
stack_contactor.requested_value	102401
stack_breaker.request_state	196610
stack_breaker.reset	196622
sc_gpo.value	94209
sc_client_id.uuid_lsb	279312
sc_client_id.uuid_msb	279313
sc_controller_wdt.expired	221184
sc_controller_heartbeat.value	278784
sc_pcs.state	279296
sc_modbus_rtu.transaction_complete	279041
sc_stackbus_packets.tx_count	77840
sc_stackbus_packets.tx_err_count	77841
sc_stackbus_packets.rx_count	77842
sc_stackbus_packets.rx_err_count	77843
sc_canbus_packets.tx_count	73984
sc_canbus_packets.tx_err_count	73985
sc_canbus_packets.rx_count	73986
sc_canbus_packets.rx_err_count	73987
sc_persist.save	18432
sc_persist.load	18433
sc_persist.error	18434
sc_aux_persist.save	18456
sc_aux_persist.load	18457
sc_aux_persist.error	18458
sc_ci_persist.save	24576
sc_ci_persist.load	24577



Register Name	Register Address
sc_ci_persist.error	24578
sc_val_persist.save	18464
sc_val_persist.load	18465
sc_val_persist.error	18466
cell_val_persist.save	18480
cell_val_persist.load	18481
cell_val_persist.error	18482
sc_locked_cfg_persist.save	18496
sc_locked_cfg_persist.load	18497
sc_locked_cfg_persist.error	18498
sc_clock_err_stats.reset	168019
sc_stackbus_stat.reset	212995
sc_stack_stat.reset	213251
sc_linkbus_stat.reset	213507
sc_rom_test_stat.reset	213763
sc_ram_test_stat.reset	214019
stack_cell_soc_stat.update	298573
stack_cell_soc_variance_stat.update	298583
stack_cell_soh_stat.update	274437
stack_balance_stat.update	281349
stack_soc.reset	90115
stack_soc.reset_soc	90116
stack_soc.reset_capacity	90117
stack_soc.measured_capacity	90125
stack_soc.adaptive_soc	90120
stack_soc.accumulative_soc	90119
cell_states.soc	293659
cell_states.vp	293658
cell_states.soc_variance	293661
cell_states.vp_variance	293660
cell_model.instantaneous_resistance	281376
cell_model.polarization_resistance	281378
cell_charge.throughput_reset	233475
cell_model.ohmic_resistance	281377
pi_op_state.requested_state	7712
pi_boot.requested_state	32
pi_fsm_stat.reset	7427

Register Name	Register Address
pi_afe_stat.reset	7939
pi_rom_test_stats.reset	7763
pi_stackbus_stat.reset	7683
pi_contactor_stats.reset	7795
pi_clock_err_stats.reset	7779
pi_aux_persist.save	4448
pi_aux_persist.load	4449
pi_aux_persist.error	4450
pi_stackbus_packets.tx_count	7808
pi_stackbus_packets.rx_count	7810
pi_stackbus_packets.tx_err_count	7809
pi_stackbus_packets.rx_err_count	7811



Configuration storage typed registers can be written to and are exempt from this rule. Refer to [Section 7.1.3, "Storage Types"](#) for information on register storage types.

## 7.3. Battery Parameters

These following settings are used to configure State-of-Charge (SoC) and State-of-Health (SoH) algorithms for operation with a particular battery chemistry.

### 7.3.1. Stack Capacity

Battery stack capacity is defined as the total amount of charge that can be extracted from a battery stack when discharging from full to empty, assuming current limits are properly followed. Nominal (or design) capacity is configured as follows:

`stack_soc.nominal_capacity`

- Nominal capacity of the battery stack
- Set to the capacity that would correspond to a full discharge



The nominal capacity of the stack is identical to that of the cells used within the stack, where a cell may be one or more electro-chemical cells directly connected in parallel. The actual capacity of the battery stack maybe less than this nominal capacity due to imbalances in SoC and capacity fade of the cells with usage. Correct configuration of the nominal capacity is essential for accurate SoC and SoH estimation.

### 7.3.2. Stack Cycle Count

In addition to the nominal capacity, the nominal cycle count for the battery stack is required for SoH

estimation based upon cycle count.

`stack_soc.nominal_cycle_count`

- Set to the expected cycle life of the battery stack assuming full discharge cycles at the configured maximum operating current
- Set to zero to ignore cycle count for SoH

### 7.3.3. State of Charge (SoC)

The state of charge of a battery is a measure of the percentage of charge contained in the battery that is available for discharge. The BMS can be configured to measure the state of charge using two different methods:

1. Accumulated SoC: Continuously measures and accumulates the charge that is charged/discharged to/from the battery.
2. Adaptive SoC: Augments the accumulated measurements with an adaptive SoC algorithm. This latter technique compensates for errors due to integration over long periods of time and for factors such as self discharge which can not be directly measured by a BMS. The algorithm requires a configuration of the [open circuit voltage](#) for the battery.

The following parameters are used to select and enable which state of charge for the Nuvation Energy BMS.

**Table 5. Current Full Condition Configuration**

Register	Type	Units	Description
<code>stack_soc.enable</code>	config	bool	Enables the state of charge measurement when set to one. Applies to both SoC techniques
<code>stack_soc.output_select</code>	config	enum	Selects which technique of state of charge is used for output on the following registers. Zero selects the accumulated SoC measurement and one selects the adaptive SoC algorithm.
<code>stack_soc.soc</code>	output	percentage	The current state of charge of the battery
<code>stack_soc.dod</code>	output	mAhr	The current depth of discharge of the battery



The adaptive state of charge algorithm does require more detailed configuration settings for proper use. Please contact [support@nuvationenergy.com](mailto:support@nuvationenergy.com) for help in configuring this algorithm for your battery chemistry.



The Sunspec Modbus point definition for SoC and DoD uses the data from the registers noted above for output.

The end limits for the state of charge (i.e. 0% and 100%) are defined by the full and empty conditions of a battery. The following subsections provide details on these end conditions and how they are configured.

#### 7.3.3.1. Battery Empty Condition

Nuvation Energy BMS supports the definition of the empty condition using either or both min/max and average cell voltages, as configured using the following registers.

#### `stack_soc.vempty`

- Defines fully discharged minimum cell voltage for operation
- Set as per application requirements, or set to zero to disable

#### `stack_soc.vemptyavg`

- Defines fully discharged average cell voltage for operation
- Set as per application requirements, or set to zero to disable

When discharging near the bottom of the battery, if the empty conditions are not met the battery will remain at 1% SoC.



For battery chemistries that do not tolerate overcharge and make use of electrical balancing circuits to keep cells balanced (e.g. Lithium-Ion), the empty conditions are typically applied against the minimum cell voltages. But for chemistries that make use of overcharge during the charging process as a way to balance the stack (e.g. lead acid), the empty conditions are typically applied against the average cell voltage.



These voltage settings must be carefully aligned with the minimum [Cell Voltage Thresholds](#) and [Stack Voltage Thresholds](#) for proper SoC operation.

### 7.3.3.2. Battery Full Condition

A battery can reach the full condition while charging (i.e. when the BMS reads negative current measurements) Nuvation Energy BMS supports the definition of the full condition using a combination of cell voltage and battery current measurements. This condition is configurable using the following register settings.

#### `stack_soc.vfull`

- Defines fully charged maximum cell voltage for operation
- Set as per application requirements, or set to zero to disable

#### `stack_soc.vfullavg`

- Defines fully charged average cell voltage for operation.
- Set as per application requirements, or set to zero to disable.

#### `stack_soc.ifull`

- Defines the largest negative current before the battery is considered full.
- Value must be negative to represent a current in the charge direction.
- Set to a large negative value (default) if this condition is not applicable.

#### `stack_charge_status.hold_current`

- Defines the upper bound for the current full condition.

- Configured as a positive value.
- Software will use the negative value of this setting for the full condition.
- Should be larger than the noise measurement of the current.

#### stack\_soc.ifull\_period

- Defines the period a current must between the prior thresholds before the full current condition is met.
- Configured as a positive value in microseconds.
- Set to a value of zero if no period is desired.

A battery is full (reaches 100% SoC) if all three of the following conditions are met:

1. The full voltage condition is met (either/both maximum or average setting).
2. The battery current is between the full current threshold and the negative hold current.
3. The battery current is maintained between the prior thresholds for the duration of the ifull period.

For example, assume  $v_{full} = 3600$  mV,  $v_{fullavg} = 0$ ,  $ifull = -1000$  mA,  $hold\_current = 5$  mA and the  $ifull\_period = 1000000$  microseconds. The full condition would be met when the first cell voltage equals or was greater than 3.6 V AND when the current was between -1000 and -5 mA for a duration of one second.

When charging at the top end of the battery, if the full conditions are not met the battery will remain at 99% SoC.



For battery chemistries that do not tolerate overcharge and make use of electrical balancing circuits to keep cells balanced (e.g. Lithium-Ion), the full conditions are typically applied against the maximum cell voltages. But for chemistries that make use of overcharge during the charging process as a way to balance the stack (e.g. lead acid), the empty conditions are typically applied against the average cell voltage.



These voltage settings must be carefully aligned with the maximum [Cell Voltage Thresholds](#) and [Stack Voltage Thresholds](#) for proper SoC operation.

## 7.3.4. Open Circuit Voltage Curve

Nuvation Energy BMS supports the use of a open-circuit voltage (OCV) table for the battery cells. The component `stack_ocv_lut` stores the voltages unique for each SoC percentage (total of 101 entries). The first table entry represents 0 percent and the last entry represents 100 percent. The voltages must be consistent with the defined the cell voltage range of [empty to full](#).

The OCV curve can be used to estimate the SoC of a stack when it is at rest. Contact [support@nuvationenergy.com](mailto:support@nuvationenergy.com) for details on how to perform this estimation.

## 7.3.5. Cell Resistance Estimation

The Nuvation Energy BMS can be configured to estimate the per cell resistance of every installed cell during charge/discharge usage. To learn more about this feature and how to configure and use it, please contact [support@nuvationenergy.com](mailto:support@nuvationenergy.com).

## 7.3.6. Nameplate Information

Nuvation Energy BMS can be configured to display nameplate information about the battery stack.

`stack_nameplate.label0`

- First 8 characters of the battery stack's name which will be displayed as a banner on the Operator Interface

`stack_nameplate.label1`

- Next 8 characters of the battery stack's name after `stack_nameplate.label0`

`stack_nameplate.label2`

- Next 8 characters of the battery stack's name after `stack_nameplate.label1`

`stack_nameplate.label3`

- Next 8 characters of the battery stack's name after `stack_nameplate.label2`

`stack_nameplate.battery_type`

- Type of battery, corresponding to Sunspec s802 point: `BatTyp`

`stack_nameplate.energy_capacity`

- The nameplate energy capacity of the battery stack in Wh, corresponding to Sunspec s801 point: `WHRtg`

`stack_nameplate.max_charge_rate`

- The nameplate maximum rate of energy transfer into the battery stack in W, corresponding to Sunspec s801 point: `WMaxChaRte`

`stack_nameplate.max_discharge_rate`

- The nameplate maximum rate of energy transfer out of the battery stack in W, corresponding to Sunspec s801 point: `WMaxDisChaRte`

`stack_nameplate.max_soc`

- The nameplate maximum state of charge expressed as a percentage, corresponding to Sunspec s801 point: `SoCnpMaxPct`

`stack_nameplate.min_soc`

- The nameplate minimum state of charge expressed as a percentage, corresponding to Sunspec s801 point: `SoCnpMinPct`

## 7.4. Stack Topology

Configuring the battery stack topology requires that the following be specified:

- Which cell voltage taps are actually connected to cells
- Which thermistor inputs are actually connected to thermistors

## 7.4.1. Cell Inputs

The Cell Interface modules may be connected to fewer cells than the maximum supported number of cell inputs.

The following registers are used to indicate which cells are actually present in the system. The index  $n$  is the zero-based index of the cell input.

`cell[n].installed`

- Indicates a cell is physically connected
- Set to 1 if connected
- Set to 0 if not connected

Cell indexing is statically assigned in such a way that `cell[0]` always refers to the first cell on the first Cell Interface in the Link Bus chain.

When using multiple Cell Interface modules, `cell[16]` refers to the first cell in the second Cell Interface in the Link Bus chain, `cell[32]` refers to the first cell in the third Cell Interface in the Link Bus chain, and so on.

## 7.4.2. Thermistor Inputs

The Cell Interface modules may be connected to fewer than the maximum supported number of thermistors. The following registers are used to indicate which thermistors are actually present in the stack. The index  $n$  is the zero-based index of the thermistor input.

`therm[n].installed`

- Indicates a thermistor is physically connected
- Set to 1 if connected
- Set to 0 if not connected

Thermistor indexing is statically assigned in such a way that `therm[0]` always refers to the first thermistor on the first Cell Interface in the Link Bus chain.

When using multiple Cell Interface modules, `therm[8]` refers to the first thermistor in the second Cell Interface in the Link Bus chain, `therm[16]` refers to the first thermistor in the third Cell Interface in the Link Bus chain, and so on.

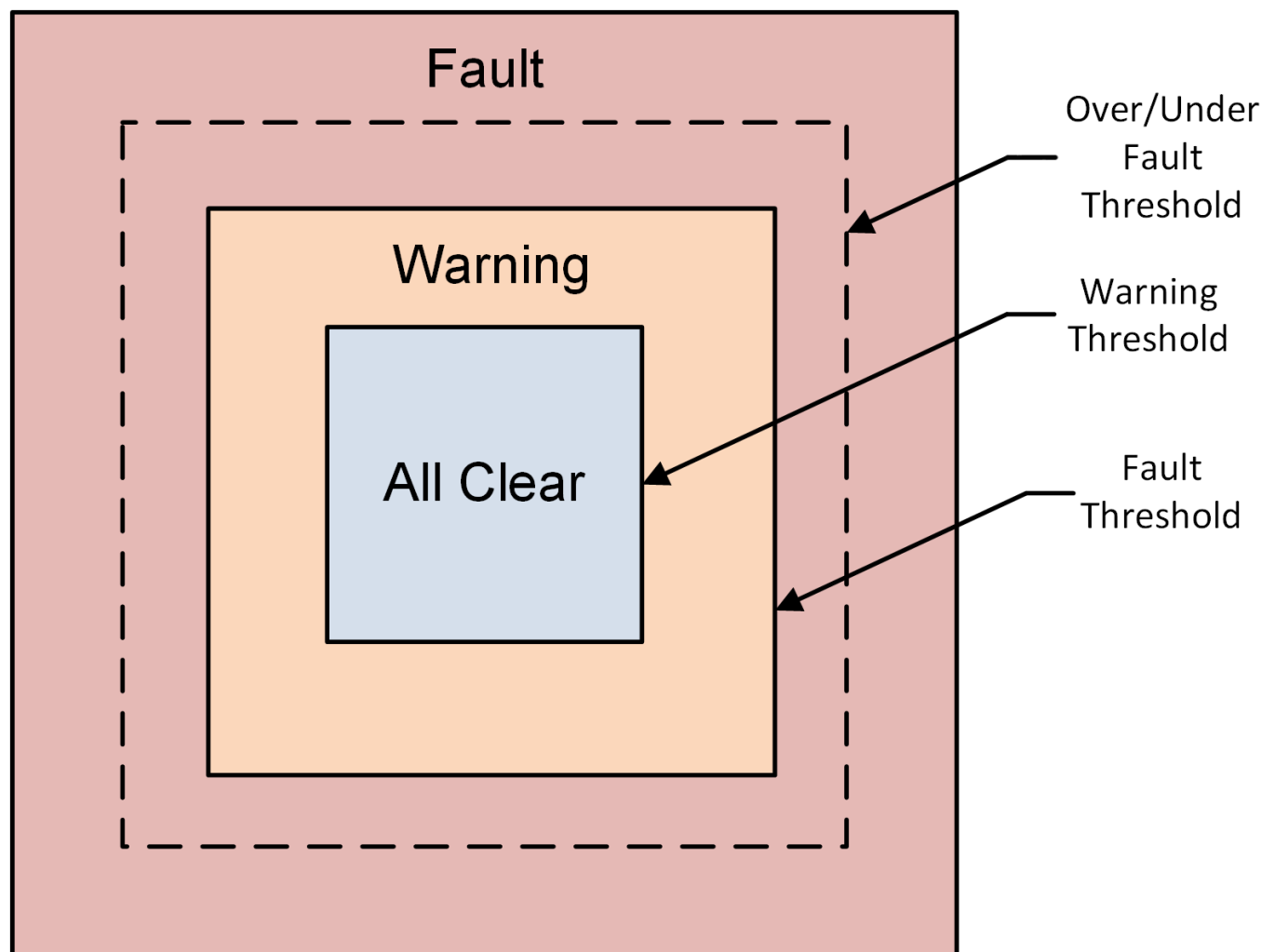
## 7.5. Operational Limits

The operational limits of a battery stack are captured in the form of voltage, temperature, and current thresholds. These thresholds must be set correctly for your battery cells and DC bus system so that the BMS can:

1. Control current during charging and discharging to keep the battery within normal operating limits.

2. Warn operators and external systems if the battery is not within normal operating limits
3. Disconnect the battery from the DC bus under a fault condition if the battery is approaching unsafe limits

Nuvation Energy BMS can either be in an operating or non-operating zone. The operating zone is also known as the *All Clear* zone while the non-operating zone is further divided into the *Warning* and *Fault* zones. All three zones are illustrated below graphically.



**Figure 68. Battery Management System Zones**

### All Clear

- The *All Clear* zone represents the subset of all system states that can be reached under normal, fully controlled operation. With proper configuration and functional control, a system should never leave this zone.

### Warning

- The *Warning* zone describes the region where the battery cells are not within their normal operating limits and Nuvation Energy BMS is taking action to try and bring the system back into the operating zone.



## Fault

- The *Fault* zone describes the region where the battery cells are approaching unsafe limits. There are two thresholds within this zone: the *Fault Threshold* and the *Over/Under Fault Threshold*. At least one of the two thresholds should be enabled.
  - At the *Fault Threshold*, Nuvation Energy BMS will gracefully disconnect the battery from the DC bus and shut down the inverter. When the battery cells return to an operable state, Nuvation Energy BMS can be configured to automatically reconnect the battery to the DC bus.
  - At the *Over/Under Fault Threshold*, Nuvation Energy BMS will disconnect the battery cells as quick as possible. An explicit user action is required to bring the system back into an operating state at this point.

## 7.5.1. Triggers

The fundamental building block used to define thresholds throughout Nuvation Energy BMS are trigger components. There are four types of trigger components, each described in the following sections.

### 7.5.1.1. Basic Triggers

A Basic Trigger is the most fundamental type of trigger in Nuvation Energy BMS. This trigger is latching meaning a tripped trigger will not self-clear when the trigger condition is cleared. The trigger will remain tripped until explicitly cleared through an external request via the Operator Interface or a supported protocol.

`trigger_name.thresh`

- The input must meet or exceed this threshold to trip the trigger.

`trigger_name.disabled`

- Set to 0 to enable the trigger
- Set to 1 to disable the trigger

### 7.5.1.2. Boolean Triggers and Standalone Triggers

Boolean and Standalone Triggers are specific types of Basic Triggers where the trigger is tripped based on either a 1 or 0 value. The two triggers are identical except for how they are tripped. Both triggers are latching. The following register is used to configure these types of triggers.

`trigger_name.disabled`

- Set to 0 to enable the trigger
- Set to 1 to disable the trigger

### 7.5.1.3. Hysteresis Triggers

A Hysteresis Trigger provides a means of hysteresis to the triggering condition. This trigger can be configured to be latching or non-latching. The following registers are used to configure this type of trigger.

`trigger_name.thresh`

- The input must meet or exceed this threshold to trip the trigger

`trigger_name.time_hyst`

- The elapsed time that the input must meet or exceed the threshold in order to trip the trigger
- Set to 0 to configure a trigger that trips instantly

`trigger_name.end_time_hyst`

- The elapsed time that the input must recover (i.e. remain below the threshold) before the trigger will clear
- Set to 0 to configure a trigger that clears instantly
- Set to > 0 to configure a trigger that clears after the defined period of time

`trigger_name.latched`

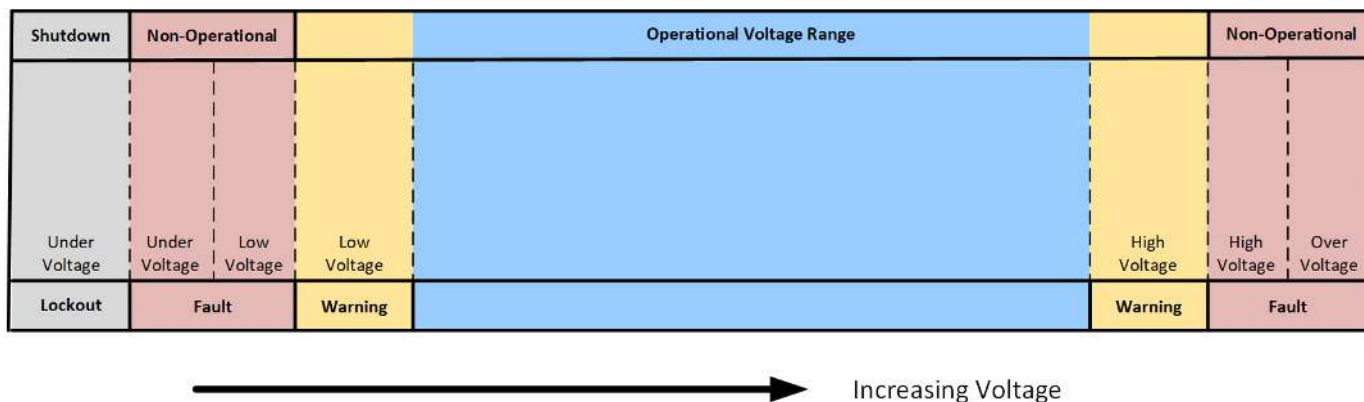
- Set to 1 for latching trigger behavior
- Set to 0 for non-latching trigger behavior

`trigger_name.disabled`

- Set to 0 to enable the trigger
- Set to 1 to disable the trigger

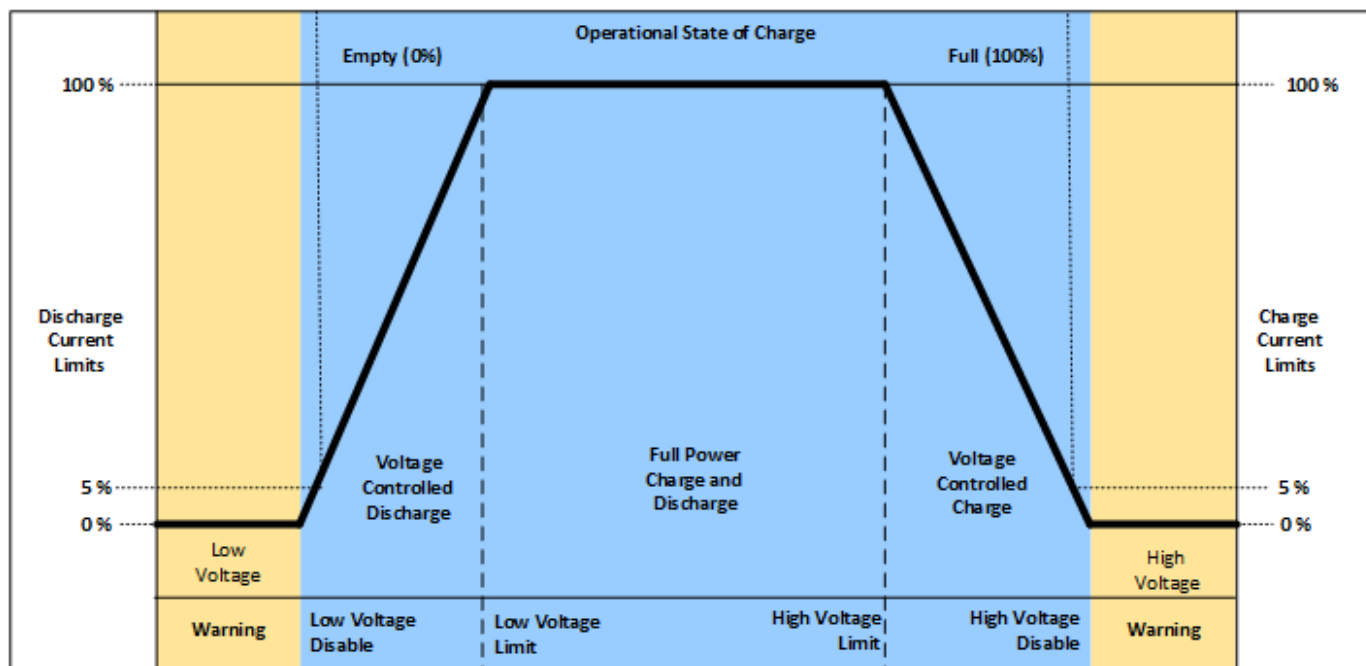
## 7.5.2. Cell Voltage Thresholds

The following diagram illustrates how cell voltage trigger thresholds are configured in typical applications.



**Figure 69. Typical Arrangement of Cell Voltage Thresholds**

These thresholds also affect the current limits of the system as shown below.



**Figure 70. Typical Arrangement of Cell Voltage Current Limits**

Most systems will make use of thresholds configured in the following order of decreasing cell voltage.

**Table 6. Cell Voltage Operational Limits**

Register	Setting
stack_fault_cell_over.thresh	The limit within the fault zone above which an explicit user operation is required to bring the system back into an operating state.
stack_fault_cell_hi.thresh	The upper limit of the warning zone as per cell specifications. Set just below stack_fault_cell_over.thresh to allow a self-clearing range within the fault zone.
stack_warn_cell_hi.thresh	The upper limit of the operating zone. Set just above stack_current_limit.voltage_cell_max.
stack_current_limit.voltage_cell_max	The voltage at which charge limits approach 0%. Set just above stack_soc.vfull.
stack_current_limit.voltage_cell_high	The voltage at which charge current limits are reduced from 100%. Set as per application requirements.
stack_current_limit.voltage_cell_low	The voltage at which discharge current limits are reduced from 100%. Set as per application requirements.
stack_current_limit.voltage_cell_min	The voltage at which discharge limits approach 0%. Set just below stack_soc.vempty.
stack_warn_cell_lo.thresh	The lower limit of the operating zone. Set just below stack_current_limit.voltage_cell_min.
stack_fault_cell_lo.thresh	The lower limit of the warning zone as per cell specifications. Set just above stack_fault_cell_under.thresh to allow a self-clearing range within the fault zone.

Register	Setting
stack_fault_cell_under.thresh	The limit within the fault zone below which an explicit user operation is required to bring the system back into an operating state.
stack_uvlo_cell_voltage.thresh	The under-voltage lockout threshold for system shutdown.



The under-voltage lockout trigger itself does not shut down the BMS when a cell voltage drops below this level. The system must have appropriate power switching hardware driven off of this trigger (e.g. through a digital output) for this shutdown to be functional. Refer to [Section 7.10.5, "Under-Voltage Shutdown"](#) for more details.

stack\_fault\_cell\_over.thresh may or may not be different from stack\_fault\_cell\_hi.thresh, depending upon system requirements. For example, if a battery system is expected to implement self-clearing (non-latching) faults during operation, the stack\_fault\_cell\_hi.thresh must be set slightly below stack\_fault\_cell\_over.thresh. This allows the stack\_fault\_cell\_hi trigger to prevent the stack\_fault\_cell\_over trigger from engaging, which would require an explicit clear action. If the stack\_fault\_cell\_over.thresh and stack\_fault\_cell\_hi.thresh are identical, it is impossible to implement self-clearing behavior on high voltage conditions. The same applies to under-voltage conditions.

The stack\_fault\_cell\_[over/under] or the stack\_fault\_cell\_[hi/lo] trigger can be disabled if only one behavior is necessary.

User-defined triggers are also available for high and low cell voltages.

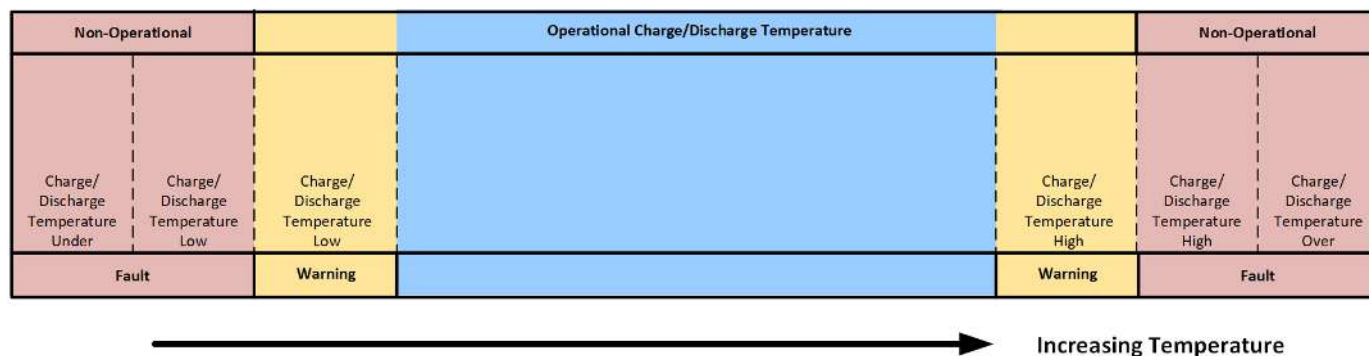
**Table 7. User-Defined Cell Voltage Triggers**

Register	Setting
stack_trig_cell_hi.thresh	User-defined high cell voltage trigger
stack_trig_cell_lo.thresh	User-defined low cell voltage trigger

## 7.5.3. Thermistor Temperature Thresholds

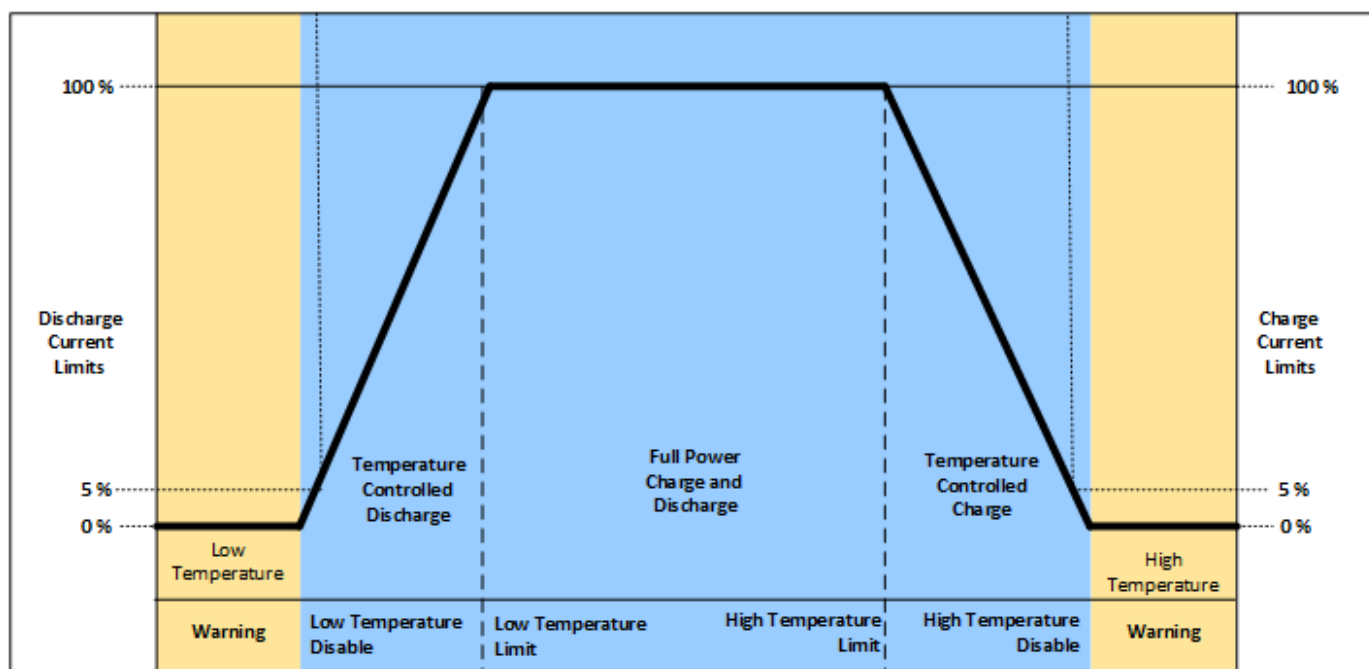
Separate configuration thresholds are provided for charging and discharging as many cells have different temperature limits in these two modes of operation. Charge triggers will only trip while the stack is charging, while discharge triggers will only trip while the stack is discharging or resting.

The following diagrams illustrates how temperature trigger thresholds are configured in typical applications.



**Figure 71. Typical Arrangement of Temperature Thresholds**

These thresholds also affect the current limits of the system as shown below.



**Figure 72. Typical Arrangement of Temperature Current Limits**

Most systems will make use of thresholds configured in the following order of decreasing thermistor temperature.

**Table 8. Charge Temperature Operational Limits**

Register	Setting
stack_fault_charge_therm_over.thresh	The limit within the fault zone above which an explicit user operation is required to bring the system back into an operating state.
stack_fault_charge_therm_hi.thresh	The upper limit of the warning charging zone as per cell specification.
stack_warn_charge_therm_hi.thresh	The upper limit of the operating charging zone. Set just above stack_current_limit.temperature_charge_max.

Register	Setting
stack_current_limit.temperature_charge_max	The temperature at which current limits approach 0% during charging.
stack_current_limit.temperature_charge_high	The temperature at which current limits are reduced from 100% during charging.
stack_current_limit.temperature_charge_low	The temperature at which current limits are reduced from 100% during charging.
stack_current_limit.temperature_charge_min	The temperature at which current limits approach 0% during charging.
stack_warn_charge_therm_lo.thresh	The lower limit of the operating charging zone. Set just below stack_current_limit.temperature_charge_min.
stack_fault_charge_therm_lo.thresh	The lower limit of the warning charging zone as per cell specification.
stack_fault_charge_therm_under.thresh	The limit within the fault zone below which an explicit user operation is required to bring the system back into an operating state.

**Table 9. Discharge Temperature Operational Limits**

Register	Setting
stack_fault_discharge_therm_over.thresh	The limit within the fault zone above which an explicit user operation is required to bring the system back into an operating state.
stack_fault_discharge_therm_hi.thresh	The upper limit of the warning discharging zone as per cell specification.
stack_warn_discharge_therm_hi.thresh	The upper limit of the operating discharging zone. Set just above stack_current_limit.temperature_discharge_max.
stack_current_limit.temperature_discharge_max	The temperature at which current limits approach 0% during discharging.
stack_current_limit.temperature_discharge_high	The temperature at which current limits are reduced from 100% during discharging.
stack_current_limit.temperature_discharge_low	The temperature at which current limits are reduced from 100% during discharging.
stack_current_limit.temperature_discharge_min	The temperature at which current limits approach 0% during discharging.
stack_warn_discharge_therm_lo.thresh	The lower limit of the operating discharging zone. Set just below stack_current_limit.temperature_discharge_min.
stack_fault_discharge_therm_lo.thresh	The lower limit of the warning discharging zone as per cell specification.
stack_fault_discharge_therm_under.thresh	The limit within the fault zone below which an explicit user operation is required to bring the system back into an operating state.

The temperature charge and discharge thresholds can be configured in a similar manner to how the cell voltage thresholds ([Section 7.5.2, "Cell Voltage Thresholds"](#)) are configured.

User-defined triggers are also available for high and low thermistor temperature during both charge and discharge.

**Table 10. User-Defined Charge Temperature Triggers**

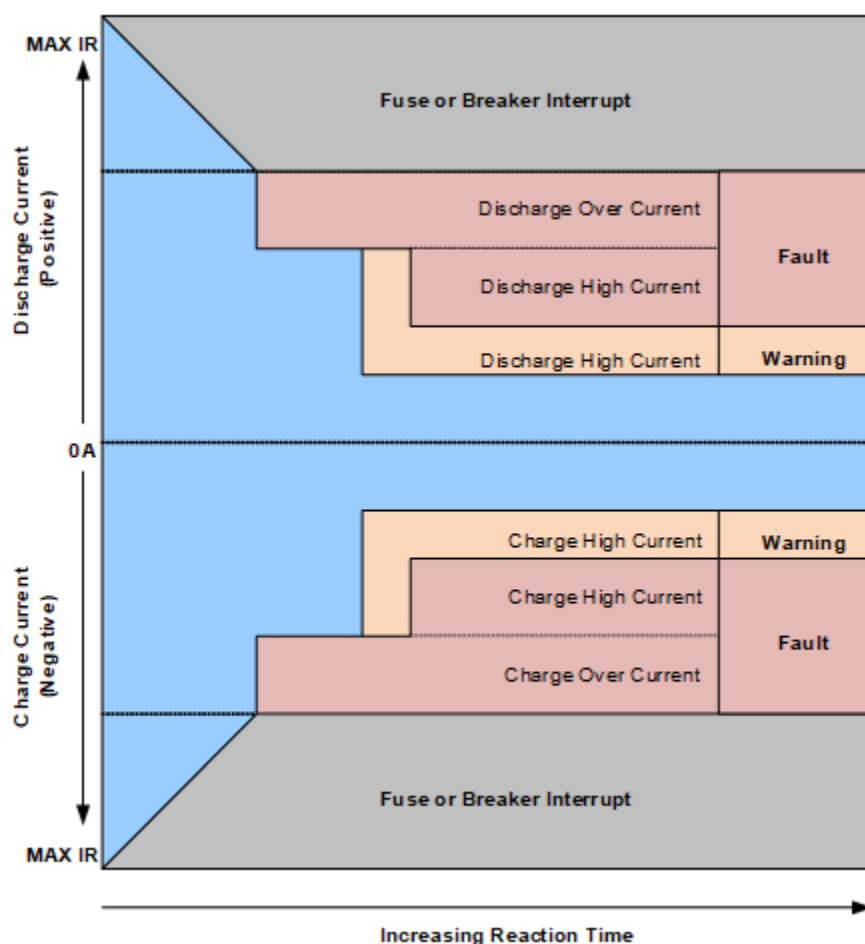
Register	Setting
stack_trig_charge_therm_hi.thresh	User-defined threshold for charge high temperature trigger.
stack_trig_charge_therm_lo.thresh	User-defined threshold for charge low temperature trigger.

**Table 11. User-Defined Discharge Temperature Triggers**

Register	Setting
stack_trig_discharge_therm_hi.thresh	User-defined threshold for discharge high temperature trigger.
stack_trig_discharge_therm_lo.thresh	User-defined threshold for discharge low temperature trigger.

## 7.5.4. Stack Current Thresholds

The following diagrams illustrate how current trigger thresholds are configured in typical applications.



**Figure 73. Typical Arrangement of Current Thresholds**

The stack current thresholds are used to define the limits of the operating zone and non-operating zone for charge and discharge currents. These limits must factor in the specifications of the battery cells as well as the limits of any DC current-carrying paths in the stack.

**Table 12. Stack Current Operational Limits**

Register	Setting
stack_fault_charge_current_over.thresh	The limit within the fault zone above which an explicit user operation is required to bring the system back into an operating state.
stack_fault_charge_current_hi.thresh	The limit of the warning charging zone as per cell specification and stack design limits.
stack_warn_charge_current_hi.thresh	The limit of the operating charging zone as per application requirements.
stack_warn_discharge_current_hi.thresh	The limit of the operating discharging zone as per application requirements.
stack_fault_discharge_current_hi.thresh	The limit of the warning discharging zone as per cell specification and stack design limits.
stack_fault_discharge_current_over.thresh	The limit within the fault zone below which an explicit user operation is required to bring the system back into an operating state.

The stack current thresholds can be configured in a similar manner to how the cell voltage thresholds ([Section 7.5.2, “Cell Voltage Thresholds”](#)) are configured. The stack\_fault\_[charge/discharge]\_current\_hi faults are typically configured with a small amount of trip time hysteresis (e.g. approximately 100–200 ms).

User-defined triggers are also available for charge and discharge currents.

**Table 13. User-Defined Stack Current Triggers**

Register	Setting
stack_trig_charge_current_hi.thresh	User-defined charge current trigger.
stack_trig_discharge_current_hi.thresh	User-defined discharge current trigger.

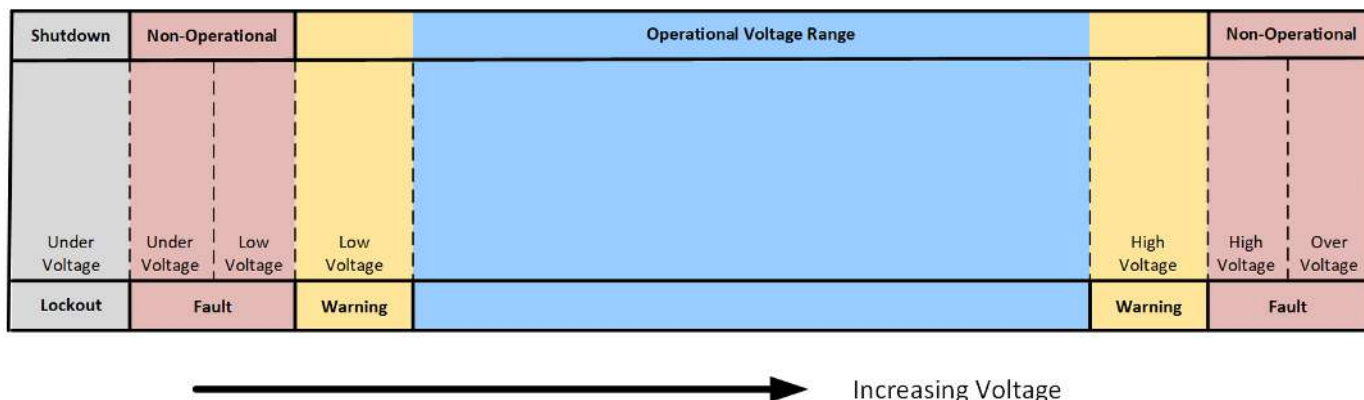


Charge current thresholds are specified as negative values and discharge current thresholds are specified as positive values.

## 7.5.5. Stack Voltage Thresholds

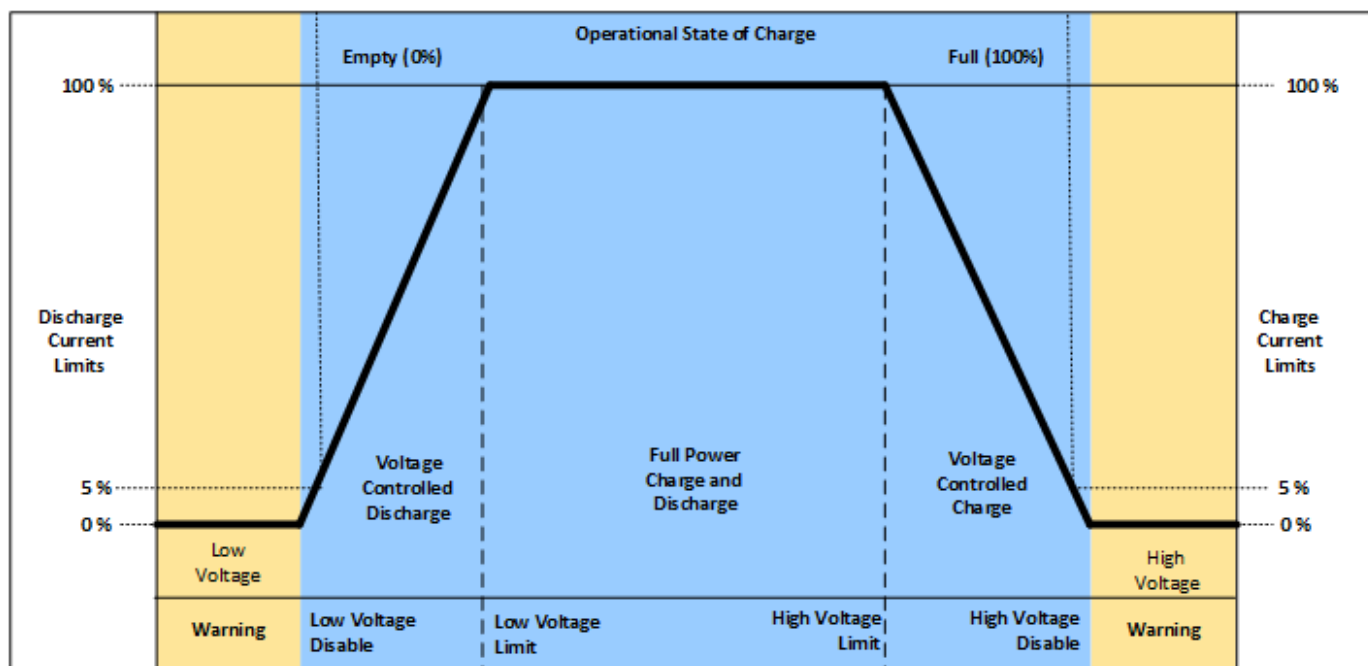
The following diagrams illustrate how stack voltage trigger thresholds are configured in typical applications.





**Figure 74. Typical Arrangement of Stack Voltage Thresholds**

These thresholds also affect the current limits of the system as shown below.



**Figure 75. Typical Arrangement of Stack Voltage Current Limits**

Stack voltage thresholds are used to define the operating voltage limits of the overall battery stack design. These thresholds are used to enforce the design limits of the battery stack or DC bus as a whole. For example, it may be necessary to limit the overall stack voltage within a certain limited range to maintain compatibility with a specific charger or inverter. Stack voltage limits also are used to ensure that average cell voltage is maintained within specified limits.

Most systems will make use of thresholds configured in the following order of decreasing stack voltage.

**Table 14. Stack Voltage Operational Limits**

Register	Setting
stack_fault_voltage_over.thresh	The limit within the fault zone below which an explicit user operation is required to bring the system back into an operating state.
stack_fault_voltage_hi.thresh	The upper limit of the warning zone as per application requirements.
stack_warn_voltage_hi.thresh	The upper limit of the operating zone. Set just above stack_current_limit.voltage_stack_max.
stack_current_limit.voltage_stack_max	The voltage at which charge limits approach 0%. Set as per application requirements.
stack_current_limit.voltage_stack_high	The voltage at which charge current limits are reduced from 100%. Set as per application requirements.
stack_current_limit.voltage_stack_low	The voltage at which discharge current limits are reduced from 100%. Set as per application requirements.
stack_current_limit.voltage_stack_min	The voltage at which discharge limits approach 0%. Set as per application requirements.
stack_warn_voltage_lo.thresh	The lower limit of the operating zone. Set just below stack_current_limit.voltage_stack_min.
stack_fault_voltage_lo.thresh	The lower limit of the warning zone as per application requirements.
stack_fault_voltage_under.thresh	The limit within the fault zone below which an explicit user operation is required to bring the system back into an operating state.
stack_uvlo_stack_voltage.thresh	The under-voltage lockout threshold for system shutdown.



The under-voltage lockout trigger itself does not shut down the BMS when the stack voltage drops below this level. The system must have appropriate power switching hardware driven off of this trigger (e.g. through a digital output) for this shutdown to be functional. Refer to [Section 7.10.5, “Under-Voltage Shutdown”](#) for more details.

The stack voltage thresholds can be configured in a similar manner to how the cell voltage thresholds ([Section 7.5.2, “Cell Voltage Thresholds”](#)) are configured.

User-defined triggers are also available for high and low stack voltages.

**Table 15. User-Defined Stack Voltage Triggers**

Register	Setting
stack_trig_voltage_hi.thresh	User-defined high stack voltage trigger.
stack_trig_voltage_lo.thresh	User-defined low stack voltage trigger.

Since stack voltage is measured independently from individual cell voltages in Nuvation Energy BMS, another important configuration threshold is the limit for mismatch between the overall stack voltage measurement and the sum of individual cell voltages.

**stack\_fault\_voltage\_sum.thresh**

- The upper limit of the operating voltage mismatch zone

- Set as per application requirements (typically at least 5–10% of `stack_fault_voltage_over.thresh`)

## 7.5.6. External Controller Heartbeat

Nuvation Energy BMS can be configured to require a heartbeat signal from an external controller in order to keep the stack online and out of fault state. A write to the MESA controller heartbeat register is expected at least once during the watchdog period. If a multi-stack controller—such as the Nuvation Energy Multi-Stack Controller—is in use, it can periodically write to this register to keep the stack out of fault state.

`sc_controller_wdt.period`

- Trip time for watchdog if heartbeat disappears
- Set to 5 seconds or as per application requirements

`sc_fault_controller_wdt.disabled`

- Set to 0 to enable controller watchdog
- Set to 1 to disable controller watchdog

If this feature is not used, the watchdog fault should be disabled.

## 7.6. Control Settings

Nuvation Energy BMS controls the current flowing through a battery stack:

1. During connection or disconnection of the battery to prevent harmful transient current events
2. During operation of a connected battery to keep the battery within its operational limits
3. During operation of a battery to keep the individual cells at a balanced state of charge
4. During a fault condition in order to protect the battery

Control for cases (1) and (4) is achieved through external switching devices that are under the control of Nuvation Energy BMS. This control is limited to hard switching. Control for case (2) is achieved through current limiting signals that are used by chargers and inverters to throttle current dynamically. Control for case (3) is implemented within Nuvation Energy BMS itself through passive balancing loads that are under control of a configurable balancing algorithm.

### 7.6.1. Stack Switch Functions

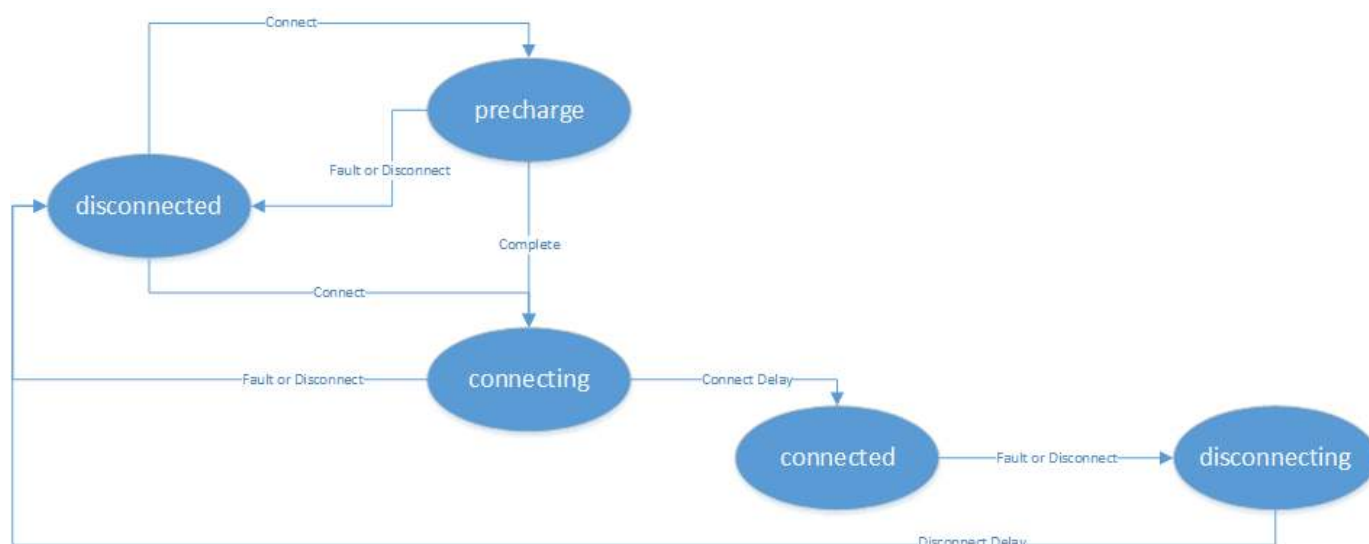
Nuvation Energy BMS defines three contactor switch functions for use within typical battery configurations:

- **Pre-charge Switch:** Connected during pre-charge operation only. Disconnected under fault condition.
- **Main Switch:** Connected after any pre-charge operation completes. Disconnected under fault condition.
- **Stack Switch:** Connected whenever either the pre-charge switch or the main switch is

connected. Disconnected under fault condition.

While a *pre-charge contactor* is optional, virtually all systems require a *main contactor* to protect the battery from unsafe conditions. A *stack contactor* is often used in conjunction with a main contactor, i.e. at the other end of the stack to isolate the battery stack completely from the DC bus and provide a level of contactor redundancy.

Stack connection and disconnection sequencing is illustrated in the state diagram below.



**Figure 76. Battery Stack States and Transitions**

As a system is connected and disconnected from the DC bus, a configurable sequencing delay is inserted before and after the connected state. During the connected state, the BMS uses current limits to control current flowing into and out of the stack. During all other states, current limits are set to zero. This allows for graceful switching behavior with no current flow under normal connect and disconnect requests.



The switch functions defined here must be mapped to appropriate outputs for use in an actual system. This process is covered in detail in [Section 7.7, "Input / Output Assignment"](#).

### 7.6.1.1. Auto-Connection Setting

It is sometimes desired for the Nuvation Energy BMS to automatically initiate a connection of a battery stack when:

- The BMS powers on
- After all faults have been cleared

This operation can be configured with the following register setting.

`stack_control.auto_connect`

- When set to a value of one, the BMS will initiate a connection of the battery stack if there are no faults triggered
- When set to a value of zero, the BMS waits for a connection request from the Operator Interface or through one of its external interface. Note a stack will not connect if there is any faults triggered.

## 7.6.1.2. Pre-Charge Switch Settings

If enabled, the pre-charge switch is engaged for a fixed (but configurable) amount of time during the pre-charge state. A bias current can be read prior to engaging the pre-charge contactor. The bias current measured is subtracted from subsequent pre-charge current measurements. If the pre-charge over-current fault is configured, the pre-charge will terminate if the fault is triggered. At the end of the pre-charge period, the stack current is compared against a maximum threshold value to determine if the pre-charge operation was successful. Upon failure, a pre-charge timeout fault is tripped. Upon successful completion, the stack connection sequence continues. During the remaining connection sequence, both the main and pre-charge contactors are engaged. At the end of the connection sequence the pre-charge contactor is de-energized.

Pre-charge behavior is configured through the following registers as required for a particular application.

### `stack_control.precharge_bias_delay`

- Determines the fixed amount of time prior to pre-charge to measure any bias current in the battery stack
- The bias current is stored in the register `stack_control.precharge_bias_current` at the end of this delay
- The bias current is removed from subsequent pre-charge current measurements before applying pre-charge fault limits.
- If enabled set to a minimum of 1 second otherwise set to 0 seconds

### `stack_control.disable_precharge`

- If set to 1 then pre-charge is disabled
- If precharge is disabled the system will skip the pre-charge step and immediately enter the next part of the connection sequence

### `stack_control.precharge_delay`

- Determines the fixed amount of time the pre-charge path is energized if enabled
- Set based upon pre-charge hardware power and thermal ratings
- This delay is limited to a minimum duration of 1 second and a maximum duration of 10 seconds



`stack_control.disable_precharge` must be set to 0 for this register to have an effect

### `stack_control.disable_connecting`

- If value is set to 1, the system will immediately enter the connected state after pre-charge completes

## `stack_control.connect_delay`

- Determines the minimum time both the pre-charge and main contactors are energized during connection
- This connection delay occurs after the pre-charge delay has successfully completed
- Set to a minimum of 1 second to ensure both contactors are energized at the end of the duration
- This delay is limited to a maximum duration of 10 seconds



`stack_control.disable_connecting` must be set to 0 for this register to have an effect

## `stack_control.precharge_max_current`

- Determines the maximum current flow at the end of `stack_control.precharge_delay` under which pre-charge can complete successfully
- Set to ensure any in-rush currents upon main switch connection are within system ratings
- Set to zero to disable pre-charge

## `stack_fault.precharge_over_current.thresh`

- Sets the over-current limit at any time during pre-charge
- If the fault is triggered, pre-charge will terminate immediately
- Limits the power dissipated of a pre-charge resistor during a short condition allowing for a smaller resistor

## `stack_fault.precharge_timeout.disabled`

- Must be set to zero if pre-charge is enabled

A pre-charge operation only completes successfully if the stack current magnitude falls below the maximum pre-charge current within the configured delay time and the `stack_fault.precharge_over_current` does not trip.



If pre-charge is enabled the fault `stack_fault.precharge_timeout` must also be enabled.

### 7.6.1.3. Sequencing Delays

The following registers are used to configure the sequencing delays used before a stack enters the connected or disconnected state.

## `stack_control.connect_delay`

- The delay between current limits engaging and the contactors connecting after a connect request
- This delay is important for systems using pre-charge
- When pre-charge is used the delay must be a minimum of one second and is limited to 10 seconds maximum



`stack_control.disable_connecting` must be set to 0 for this register to have an effect,

otherwise the system will behave as if this delay is 0s

#### `stack_control.disconnect_delay`

- The delay between current limits disengaging and the contactors disconnecting after a disconnect request
- This delay is important to allow a PCS time to ramp down before the BMS opens the contactors
- This delay should not be less than one second and is limited to 10 seconds maximum

The disconnect switching delay only applies to disconnect requests. Under a fault condition, the stack is disconnected immediately.

## 7.6.2. Stack Current Limits

### 7.6.2.1. Maximum Operating Currents

The maximum continuous operating charge and discharge currents must be configured for current limiting to function properly. These values correspond to the current limit values that will be used during normal wide-open operation (i.e. no throttling).

#### `stack_current_limit.max_charge_current`

- Magnitude of maximum continuous operating charge current

#### `stack_current_limit.max_discharge_current`

- Magnitude of maximum continuous operating discharge current



The current limits given above are magnitudes only—i.e. both charge and discharge current limits are positive.

### 7.6.2.2. Minimum Charge Current

The minimum charge current is the constant charging current that should be applied as the battery reaches the end of its charge cycle. The BMS will ensure that the charge current limit does not fall below this minimum value until the battery has reached its maximum charging voltage.

#### `stack_current_limit.min_charge_current`

- Minimum charge current to be applied at the end of charge cycle
- Set as per battery manufacturer recommendations (typically below C/20)

This setting must be configured in conjunction with the full thresholds defined by [Stack Capacity](#). For example, the stack can be considered full once the current limit reaches the minimum charge current.



The current limit given above is magnitude only—i.e. charge current limit is positive.

### 7.6.2.3. Current Limiting Response Times

The current limiting control loop can be tuned for stable and responsive behavior in a variety of systems. Two independent settling times are provided to allow independent adjustment of the response to decreases in current limits (attack time) and increases in current limits (decay time).

`stack_current_limit.attack_settling_time`

- Settling time for decreases in current limits
- Typically this is between 1 and 5 seconds

`stack_current_limit.decay_settling_time`

- Settling time for increases in current limits
- Typically this is on the order of 10x larger than `stack_current_limit.attack_settling_time`

`stack_current_limit.nonlinear_rolloff`

- Enables a non-linear roll-off of the current limits when set to True (i.e. a value of one)
- Current limits will follow a linear response when set to False (i.e. a value of zero)

Since the attack time determines how quickly the current limits can respond before a potential fault conditions opens a switch, it is critical to have sufficient control bandwidth here to avoid tripping faults.

A non-zero settling time is critical in most applications to avoid oscillations in the presence of noise and other imperfections in high-power control of the DC current in the charger and/or inverter.

When the nonlinear roll-off is enabled, the slope of the current limits at the end of the charge/discharge regions will be smaller (closer to zero) than the linear response. A nonlinear roll-off will reduce large changes in variables (such as the cell voltage) on the current limit output.

## 7.6.3. Passive Cell Balancing

When multiple cells are connected in series to form a larger battery stack, it is important to ensure each cell is giving equal contributions to the system. The effects of a single low State-of-Charge or a single high State-of-Charge cell will dominate the performance of the large battery stack. The act of equalizing State-of-Charge of multiple series-connected cells is called *balancing* and there are many types of balancing.

Nuvation Energy BMS implements a *passive* balancing solution. Cells with high State-of-Charge are discharged via internal bleed resistors which are enabled on a per-cell basis. Properly adjusting the algorithm settings for your cells is necessary to achieve a well-performing system.



Balancing is not supported on Cell Interface - 12 V 4 channel.

A number of configurable settings are used to fine-tune the passive balancing algorithm for voltage, temperature, current, and duty cycle.

The balancer must be enabled for balancing to take place.

`stack_cell_balancer.enabled`

- Enables or disables balancing operation



- Set to 1 to enable
- Set to 0 to disable

## 7.6.3.1. Cell Balancing Voltage Settings

Both absolute and relative cell voltage thresholds are used to safely balance a stack of batteries.

**min\_enable\_voltage** is the absolute threshold that determines the voltage below which a cell will not be balanced. This prevents over discharging in a system even with large imbalances.

**voltage\_delta** is the relative threshold that is used to determine when a system is balanced. Balancing for a given cell will take place when that cell's voltage is at least **voltage\_delta** millivolts higher than the minimum cell voltage.

**stack\_cell\_balancer.min\_enable\_voltage**

- Minimum voltage threshold for balancing (mV). The minimum cell voltage threshold for balancing must be greater than, or equal to, 1.1 V.
- Typically this is set higher than **stack\_current\_limit.voltage\_cell\_high**

**stack\_cell\_balancer.voltage\_delta**

- Delta voltage threshold for balancing (mV)
- Delta between a given cell's voltage and the minimum cell voltage
- Typically this is set between 5 and 25 mV



If **stack\_cell\_balancer.voltage\_delta** is set to zero, the system will continue balancing down all cells (even if the difference between min and max is zero) until they reach **stack\_cell\_balancer.min\_enable\_voltage**. This mode can be used to passively balance all cells in a stack to a specific open-circuit voltage.

## 7.6.3.2. Cell Balancing Temperature Settings

Temperature limits are used to prevent the cells and the BMS module from overheating. If the following temperature threshold is exceeded, balancing is disabled for all cells in the stack.

**stack\_cell\_balancer.max\_enable\_temperature**

- The upper cell temperature limit, in degrees °C, as measured by thermistors external to the BMS
- Set per application requirements.

## 7.6.3.3. Cell Balancing Current Settings

Thresholds based on stack current allow the balancer to be fine-tuned to run during specific portions of the charge and discharge cycle.

**stack\_cell\_balancer.min\_enable\_current**

- The minimum current at which balancing remains enabled (mA)

- This is typically set to a negative value to enable balancing below certain charge currents
- For example, a value of -10000 causes balancing to be **disabled** when charging at greater than 10 A

`stack_cell_balancer.max_enable_current`

- The maximum current at which balancing remains enabled (mA)
- This may be a negative or positive value depending upon application requirements
- For example, a value of 200 causes balancing to be **disabled** when discharging at greater than 200 mA

The two most common use cases are:

1. **Balance only while charging:** In this case, both the minimum and maximum current thresholds are set to negative values that correspond to the range of charge currents under which balancing should take place.
2. **Balance while charging and holding:** In this case, the maximum current threshold is set to a slightly positive value so that the stack will balance when it is idle or disconnected. The level of discharge current flow tolerated during idle balancing is application-specific and is thus configurable.

#### 7.6.3.4. Cell Balancing Duty Cycle

All the energy that is removed from a cell by passive balancing is dissipated as heat. If the amount of heat dissipated is too great, its desirable to reduce the amount of balancing applied to the cells. Passive balancing is applied for the majority of a cell voltage scan period defined by the register `sc_linkbus[0].scan_period`. Balancing is disabled every linkbus scan cycle to measure the cell voltages. The register `sc_linkbus[0].balance_settle_period` provides the period of time balancing is turned off before measuring cell voltages. The duration of balancing applied to all cells can be reduced by increasing this delay period.



The `sc_linkbus[0].balance_settle_period` register value must be between a minimum of 50 ms and must be smaller than the linkbus scan period.

#### 7.6.4. Breaker Interruption

The Nuvation Energy BMS can be configured to accept an external input to open the contactors via a fault. The input is provided through a [General Purpose Digital Input](#). The GPI is directed via configuration to a trigger which can generate a fault forcing the contactors to open. The following table provides the configuration to enable an externally triggered breaker interruption.

Register	Type	Description	Value
<code>stack_breaker[0].enable</code>	Bool	Enables the breaker operation	1
<code>stack_breaker[0].tripped_address_enable</code>	Bool	Enables the forwarding of data to the breaker	1
<code>stack_breaker[0].tripped_address</code>	Address	Address of the GPI which to read from	Refer to <a href="#">address config</a>

Register	Type	Description	Value
stack_fault_breaker_tripped[0].disabled	Bool	Enable the breaker fault	1
General trigger configuration		Follow general <a href="#">trigger configuration</a> to desired operation	

## 7.7. Input / Output Assignment

The Stack Controller implements the following inputs and outputs:

- Fault Pilot output signal (sent on the Stack Bus)
- Four general purpose digital outputs
- Four general purpose digital inputs

The Fault Pilot signaling mechanism is a dedicated hardware signaling path between the Stack Controller and Power Interface that is used to rapidly open the contactors in the case of a fault condition or processor failure. It is configurable for advanced applications that may require customized behavior ([Section 7.10.4, "Fault Pilot Signal"](#)).

The Power Interface implements the following inputs and outputs:

- Fault Pilot input signal (received from the Stack Bus)
- Four contactor output drivers

Contactor outputs and general purpose digital inputs and outputs present on Nuvation Energy BMS are implemented to allow for assignment of pin functions through configuration rather than through hard-wired implementation. This means that the pins connected to any external contactors, switches, or other digital inputs or outputs must be mapped in configuration to the appropriate BMS function for that system.

### 7.7.1. Contactor Outputs

Contactor output drivers are assigned to stack switching functions through configuration. Contactors are also configured as directional or non-directional. A directional contactor has a preferred direction for breaking current. Nuvation Energy BMS will open any non-directional contactors or directional contactors aligned with stack current flow first. Directional contactors that are opposed to stack current flow will be opened after a configurable delay. An optional feedback mechanism can be configured for contactors with a feedback line. Once a contactor's feedback output is wired into a GPI and configured, if the contactor fails to open/close, the feedback line will indicate the problem and the BMS can flag a fault.

The following registers are used to configure contactor outputs. The index **n** is the zero-based index of the hardware contactor coil output.

`stack_contactor[n].address_enabled`

- Set to 1 to enable the contactor function specified in `stack_contactor[n].address`

`stack_contactor[n].inverted`

- When set to 0, contactor is **energized** when assigned function value is 1
- When set to 1, contactor is **energized** when assigned function value is 0

`stack_contactor[n].address`

- Determines the function mapped to the contactor
- Set to `@stack_control.main_switch_stateto` function as main switch
- Set to `@stack_control.stack_switch_stateto` function as stack switch
- Set to `@stack_control.precharge_switch_stateto` function as pre-charge switch

`stack_contactor[n].direction`

- Set to 0 for a non-directional contactor that breaks any current
- Set to 1 for a directional contactor that breaks charge current
- Set to 2 for a directional contactor that breaks discharge current

`stack_contactor[n].delay`

- Time to wait before opening the contactor in the non-preferred direction
- When directional contactors are used, set this delay to a minimum of 1 second, otherwise set the delay to 0 seconds

`stack_contactor[n].installed`

- Set to 1 to indicate a contactor is physically wired into the BMS
- Only installed contactors will flag hardware failures via the `stack_contactor[n].coil_error` or `stack_contactor[n].feedback_error`

`stack_contactor[n].feedback_enable`

- Set to 1 to enable contactor feedback from a configured input
- State mismatches between `stack_contactor[n].feedback_value` and `stack_contactor[n].value` will result in `stack_contactor[n].feedback_error` being set to 1

For advanced applications, contactor outputs may be configured to be driven from any Boolean register within Nuvation Energy BMS.

## 7.7.2. General Purpose Digital Outputs

The most commonly used *General Purpose Digital Output* functions are:

- **Charge current disable:** A control signal that is asserted when charging should be disabled
- **Discharge current disable:** A control signal that is asserted when discharging should be disabled
- **Fault state:** A signal that is asserted when faults are present within the system
- **Trigger state:** A trigger signal for external devices that is asserted when a specific trigger within the system is tripped

*General Purpose Digital Output* functions are assigned through the following configuration registers.

The index **n** is the zero-based index of the *General Purpose Digital Output* hardware pin.

`sc_gpo[n].address_enabled`

- Set this to 1 to enable the function specified in `sc_gpo[n].address`

`sc_gpo[n].inverted`

- When set to 0, GPO output switch is **closed** when assigned function value is 0
- When set to 1, GPO output switch is **closed** when assigned function value is 1

`sc_gpo[n].address`

- Determines the function mapped to output

The configuration settings for the most common functions are illustrated in the table below.

**Table 16. Common Assignments for General Purpose Digital Output**

Function	<code>sc_gpo[n].address</code>	<code>sc_gpo[n].inverted</code>
Charge Current Enable	@stack_current_limit.charge_current_disable	0
Discharge Current Enable	@stack_current_limit.discharge_current_disable	0
Fault State	@stack_trigger_summary.fault	1
Trigger State	@trigger_name.trig	1

For advanced applications, *General Purpose Digital Outputs* may be configured to be driven from any Boolean register within the BMS.



The Power Interface also has GPOs via the `pi_gpo` but these should not be configured for general usage.

## 7.7.3. General Purpose Digital Inputs

The most commonly used *General Purpose Digital Input* functions are:

- **Clear faults:** Hardware input to clear any latched fault conditions
- **Connect request:** Request to connect the battery stack to the DC bus
- **Auxiliary contactor feedback:** Provides feedback to compare an auxiliary contactor state with current contactor state.

*General Purpose Digital Input* functions are assigned through the following configuration registers. The index **n** is the zero-based index of the *General Purpose Digital Input* hardware pin.

`sc_gpi[n].address_enabled`

- When set to 1, the state of the input pin is propagated to the destination register address

`sc_gpi[n].inverted`

- When set to 0, GPI input value is 1 if hardware GPI is asserted
- When set to 1, GPI input value is 0 if hardware GPI is asserted

`sc_gpi[n].address`

- The destination register address to populate with the state of the input pin

`sc_gpi[n].rising_edge_triggered`

- When set to 1, the input value will be populated to the destination upon detection of a rising edge

`sc_gpi[n].falling_edge_triggered`

- When set to 1, the input value will be populated to the destination upon detection of a falling edge

If the *General Purpose Digital Input* is configured as neither rising nor falling edge triggered, the input value is continuously populated into the destination address. The configuration settings for the most common functions are illustrated in the table below.

**Table 17. Common Assignments for General Purpose Digital Input**

Function	<code>sc_gpi[n].address</code>	<code>sc_gpi[n].inverted</code>	<code>sc_gpi[n].rising_edge_triggered</code>
Clear Faults	<code>@stack_trigger_summary.clear_faults</code>	0	1
Connect Request	<code>@stack_control.requested_state</code>	0	1
Contactors Feedback	<code>@stack_contactor[m].feedback_value</code>	0	0

For advanced applications, *General Purpose Digital Input* may be configured to drive any Boolean register within the BMS.

## 7.8. Protocol Settings

The High-Voltage BMS supports the following interfaces (on the Stack Controller module) for connection with external systems:

- 10/100 Ethernet for Modbus TCP and Operator Interface connectivity
- CAN Bus (not available in no-CAN variants (\*-NC))
- RS485 for Modbus RTU

### 7.8.1. CAN Bus



The details below do not apply to no-CAN variants (\*-NC) as they do not have a CAN bus interface.

Nuvation Energy BMS uses a flexible CAN reporting implementation which maps BMS registers to CAN message identifiers.

Refer to [Section 8.2, “CAN Bus Protocol Support”](#) for more information on using this interface.

Up to 64 individual registers may be configured for periodic reporting by the BMS. Additionally, 4 configurable bulk report blocks are available for reporting repeating blocks of registers such as cell voltage and temperature.

The basic CAN configuration can be done with the components and registers described below.

#### `sc_canbus.enabled`

- A flag which enables the CAN bus interface
- Set to 1 to enable CAN reporting

#### `sc_canbus.base_can_address`

- The base CAN bus message ID
- Messages are assigned sequential IDs starting at this value



All reports starting at the base ID must fit into the 11-bit CAN bus message ID.

#### `sc_canbus.report_interval`

- The periodic reporting period for CAN message broadcasts

#### `sc_canbus.report_msg_interval`

- The inter-message delay interval for outgoing CAN messages This register should not be set to zero to allow [fair sharing](#) of the CAN bus with other devices.

#### `sc_canbus_packets.err_rate_window`

- The time window to average communication errors over when calculating the error rate

The standard configuration uses the following CAN reporting base settings:

**Table 18. Standard Configuration for CAN Reporting**

Register	Setting	Note
<code>sc_canbus.enable</code>	1	Set to 1 to enable CAN reports.
<code>sc_canbus.base_can_address</code>	0x100	
<code>sc_canbus.report_interval</code>	500000	
<code>sc_canbus.report_msg_interval</code>	1000	See <a href="#">Section 7.8.1.3, "Inter Report Message Interval"</a>
<code>sc_canbus_packets.err_rate_window</code>	30000000	

## 7.8.1.1. Individual Register Mapping

Nuvation Energy BMS has a standard set of mapped registers that covers common use cases suitable for most systems. The addresses of the 64 registers associated with this reporting are configured in the following registers.

#### `sc_canbus_map[0:63].address`

- The register address of a value to map over CAN bus. A value of 0 disables the associated message.

`sc_canbus_map[0:63].command`

- Set to 1 to enable *command messages* for the associated register. Set to 0 to enable *reported messages* for the associated register.

These mapped messages are ordered and will have sequential CAN IDs starting at `sc_canbus.base_can_address`.

A standard configuration for CAN messages of individual registers uses the following settings:

**Table 19. Standard Configuration for Individual Register CAN Mapping**

Register	Setting
<code>sc_canbus_map[0].address</code>	<code>@sc_clock.seconds</code>
<code>sc_canbus_map[1].address</code>	<code>@stack_power.voltage</code>
<code>sc_canbus_map[2].address</code>	<code>@stack_power.current</code>
<code>sc_canbus_map[3].address</code>	<code>@stack_soc.soc</code>
<code>sc_canbus_map[4].address</code>	<code>@stack_soc.dod</code>
<code>sc_canbus_map[5].address</code>	<code>@stack_cell_stat.max</code>
<code>sc_canbus_map[6].address</code>	<code>@stack_cell_stat.min</code>
<code>sc_canbus_map[7].address</code>	<code>@stack_cell_stat.avg</code>
<code>sc_canbus_map[8].address</code>	<code>@stack_therm_stat.max</code>
<code>sc_canbus_map[9].address</code>	<code>@stack_therm_stat.min</code>
<code>sc_canbus_map[10].address</code>	<code>@stack_therm_stat.avg</code>
<code>sc_canbus_map[11].address</code>	<code>@stack_trigger_summary.no_faults</code>
<code>sc_canbus_map[12].address</code>	<code>@stack_current_limit.charge_current_disable</code>
<code>sc_canbus_map[13].address</code>	<code>@stack_current_limit.discharge_current_disable</code>
<code>sc_canbus_map[14].address</code>	<code>@stack_current_limit.charge_current_limit</code>
<code>sc_canbus_map[15].address</code>	<code>@stack_current_limit.charge_current_percent</code>
<code>sc_canbus_map[16].address</code>	<code>@stack_current_limit.discharge_current_limit</code>
<code>sc_canbus_map[17].address</code>	<code>@stack_current_limit.discharge_current_percent</code>
<code>sc_canbus_map[18].address</code>	<code>@stack_control.connection_state</code>
<code>sc_canbus_map[19].address</code>	<code>@stack_trigger_summary.clear_faults</code>
<code>sc_canbus_map[20].address</code>	<code>@stack_control.requested_state</code>
<code>sc_canbus_map[21].address</code>	<code>@sc_controller_heartbeat.value</code>
<code>sc_canbus_map[22:63].address</code>	0
<code>sc_canbus_map[0:18].command</code>	0
<code>sc_canbus_map[19:21].command</code>	1
<code>sc_canbus_map[22:63].command</code>	0

With `sc_canbus.base_can_address = 0x100`, the above configuration would result in the following CAN message IDs:



**Table 20. CAN IDs for Individual Registers Using the Standard Configuration**

CAN ID	Message	Unit
0x100	Clock	Seconds
0x101	Stack Voltage	mV
0x102	Stack Current	mA
0x103	State of Charge	%
0x104	Depth of Discharge	mAh
0x105	Maximum Cell Voltage	mV
0x106	Minimum Cell Voltage	mV
0x107	Average Cell Voltage	mV
0x108	Maximum Temperature	°C
0x109	Minimum Temperature	°C
0x10A	Average Temperature	°C
0x10B	Overall Safe	Boolean
0x10C	Safe to Charge	Boolean
0x10D	Safe to Discharge	Boolean
0x10E	Charge Current Limit	mA
0x10F	Charge Percent Limit	%
0x110	Discharge Current Limit	mA
0x111	Discharge Percent Limit	%
0x112	Stack Control Connection State	Enumeration

The enumerated values for Stack Control Connection State are as follows

- 0 = Disconnected
- 1 = Precharge
- 2 = Connecting
- 3 = Connected
- 4 = Disconnecting

## 7.8.1.2. Bulk Register Reporting

In addition to individual register reporting, four configurable bulk report blocks are available for reporting repeating blocks of registers such as cell voltage and temperature.

From the receiver's point of view, there is no difference between a message for individual registers and a message for bulk registers. The main difference is in the configuration.

Repeating blocks of CAN bus messages are configured using the following registers.

`sc_canbus_bulk_map[0:3].baseaddress`

- The register address from which to start bulk reading
- A value of 0 disables the associated messages from being broadcast

`sc_canbus_bulk_map[0:3].baseenableaddress`

- The register address used to enable transmission of a CAN message

`sc_canbus_bulk_map[0:3].offset`

- The offset to add to the base addresses between each read
- This should equal the number of registers in the component being mapped
- For example, if the component `stack_cell_soh` has 2 registers (soh and service) then an offset of 2 would mean that only soh is reported for each cell, skipping over service

`sc_canbus_bulk_map[0:3].numtoread`

- The number of registers to read and report in total for this bulk report



The addresses used for `baseaddress` and `baseenableaddress` need to be based from the same component such as `cell` or `therm`.

The messages for bulk reports are ordered and have sequential CAN IDs. The first bulk report (associated with `sc_canbus_bulk_map[0]`) uses CAN IDs starting at `sc_canbus.base_can_address + 64`.

The next bulk report (associated with `sc_canbus_bulk_map[1]`) uses CAN IDs starting at `(sc_canbus.base_can_address + 64 + sc_canbus_bulk_map[0].numtoread)`.

Changing `sc_canbus_bulk_map[n].numtoread` shifts the CAN IDs of the subsequent bulk reports (`sc_canbus_bulk_map[n+1]`, ..., `sc_canbus_bulk_map[3]`).

A standard configuration for CAN reporting of bulk registers uses the following settings (for a system with only one set of cells and thermistors):

**Table 21. Standard Configuration for Bulk Register CAN Reporting**

Register	Setting
<code>sc_canbus_bulk_map[0].baseaddress</code>	<code>@cell.voltage</code>
<code>sc_canbus_bulk_map[0].baseenableaddress</code>	<code>@cell.installed</code>
<code>sc_canbus_bulk_map[0].offset</code>	6
<code>sc_canbus_bulk_map[0].numtoread</code>	16
<code>sc_canbus_bulk_map[1].baseaddress</code>	<code>@therm.temperature</code>
<code>sc_canbus_bulk_map[1].baseenableaddress</code>	<code>@therm.installed</code>
<code>sc_canbus_bulk_map[1].offset</code>	4
<code>sc_canbus_bulk_map[1].numtoread</code>	8
<code>sc_canbus_bulk_map[2:3].baseaddress</code>	0
<code>sc_canbus_bulk_map[2:3].baseenableaddress</code>	0
<code>sc_canbus_bulk_map[2:3].offset</code>	0
<code>sc_canbus_bulk_map[2:3].numtoread</code>	0

With `sc_canbus.base_can_address = 0x100`, the above configuration would result in the following CAN message IDs:

**Table 22. CAN IDs for Bulk Register Using the Standard Configuration**

CAN ID	Message	Unit
0x140	Cell 0 Voltage	mV
0x141	Cell 1 Voltage	mV
0x142	Cell 2 Voltage	mV
0x143	Cell 3 Voltage	mV
0x144	Cell 4 Voltage	mV
0x145	Cell 5 Voltage	mV
0x146	Cell 6 Voltage	mV
0x147	Cell 7 Voltage	mV
0x148	Cell 8 Voltage	mV
0x149	Cell 9 Voltage	mV
0x14A	Cell 10 Voltage	mV
0x14B	Cell 11 Voltage	mV
0x14C	Cell 12 Voltage	mV
0x14D	Cell 13 Voltage	mV
0x14E	Cell 14 Voltage	mV
0x14F	Cell 15 Voltage	mV
0x150	Thermistor 0 Temperature	°C
0x151	Thermistor 1 Temperature	°C
0x152	Thermistor 2 Temperature	°C
0x153	Thermistor 3 Temperature	°C
0x154	Thermistor 4 Temperature	°C
0x155	Thermistor 5 Temperature	°C
0x156	Thermistor 6 Temperature	°C
0x157	Thermistor 7 Temperature	°C

Some repeating blocks of registers may have instances whose addresses are not immediately adjacent, due to provisioning of registers for systems with a higher channel count.

For example, in a system with two sets of 12 cells, each set of registers may span 16 register blocks. In this scenario, `sc_canbus_bulk_map[0:3].numtoread` should be set to 32, not 24. `sc_canbus_bulk_map[0:3].numtoread` should be a multiple of 16 for bulk registers associated with cells regardless of `cell[0:799].installed`.

`sc_canbus_bulk_map[0:3].numtoread` should be a multiple of 8 for bulk registers associated with thermistor regardless of `therm[0:399].installed`.

A system with two sets of 12 cells and two sets of 8 thermistors would have

sc\_canbus\_bulk\_map[0].numtoread = 32 and sc\_canbus\_bulk\_map[1].numtoread = 16.

With this modification, the above configuration would result in the following CAN message IDs:

**Table 23. CAN IDs for Bulk Register Using the Standard Configuration**

CAN ID	Message	Unit	Note
0x140	Cell 0 Voltage	mV	1st cell of 1st set
0x141	Cell 1 Voltage	mV	2nd cell of 1st set
0x142	Cell 2 Voltage	mV	3rd cell of 1st set
0x143	Cell 3 Voltage	mV	4th cell of 1st set
0x144	Cell 4 Voltage	mV	5th cell of 1st set
0x145	Cell 5 Voltage	mV	6th cell of 1st set
0x146	Cell 6 Voltage	mV	7th cell of 1st set
0x147	Cell 7 Voltage	mV	8th cell of 1st set
0x148	Cell 8 Voltage	mV	9th cell of 1st set
0x149	Cell 9 Voltage	mV	10th cell of 1st set
0x14A	Cell 10 Voltage	mV	11th cell of 1st set
0x14B	Cell 11 Voltage	mV	12th cell of 1st set
0x150	Cell 16 Voltage	mV	1st cell of 2nd set
0x151	Cell 17 Voltage	mV	2nd cell of 2nd set
0x152	Cell 18 Voltage	mV	3rd cell of 2nd set
0x153	Cell 19 Voltage	mV	4th cell of 2nd set
0x154	Cell 20 Voltage	mV	5th cell of 2nd set
0x155	Cell 21 Voltage	mV	6th cell of 2nd set
0x156	Cell 22 Voltage	mV	7th cell of 2nd set
0x157	Cell 23 Voltage	mV	8th cell of 2nd set
0x158	Cell 24 Voltage	mV	9th cell of 2nd set
0x159	Cell 25 Voltage	mV	10th cell of 2nd set
0x15A	Cell 26 Voltage	mV	11th cell of 2nd set
0x15B	Cell 27 Voltage	mV	12th cell of 2nd set
0x160	Thermistor 0 Temperature	°C	1st thermistor of 1st set
0x161	Thermistor 1 Temperature	°C	2nd thermistor of 1st set
0x162	Thermistor 2 Temperature	°C	3rd thermistor of 1st set
0x163	Thermistor 3 Temperature	°C	4th thermistor of 1st set
0x164	Thermistor 4 Temperature	°C	5th thermistor of 1st set
0x165	Thermistor 5 Temperature	°C	6th thermistor of 1st set
0x166	Thermistor 6 Temperature	°C	7th thermistor of 1st set
0x167	Thermistor 7 Temperature	°C	8th thermistor of 1st set
0x168	Thermistor 8 Temperature	°C	1st thermistor of 2nd set

CAN ID	Message	Unit	Note
0x169	Thermistor 9 Temperature	°C	2nd thermistor of 2nd set
0x16A	Thermistor 10 Temperature	°C	3rd thermistor of 2nd set
0x16B	Thermistor 11 Temperature	°C	4th thermistor of 2nd set
0x16C	Thermistor 12 Temperature	°C	5th thermistor of 2nd set
0x16D	Thermistor 13 Temperature	°C	6th thermistor of 2nd set
0x16E	Thermistor 14 Temperature	°C	7th thermistor of 2nd set
0x16F	Thermistor 15 Temperature	°C	8th thermistor of 2nd set

### 7.8.1.3. Inter Report Message Interval

By default the CAN report messages are sent successively one after another. Some CAN receivers will experience issues receiving CAN messages at this maximum rate. The Battery Management System can be configured to inject a pause between report messages. This delay is configured using the register `sc_canbus[0].report_msg_interval`. The value of this register represents the number of microseconds inserted between CAN report messages. This delay applies to all CAN report messages. A delay inserted between report messages will also allow better CAN bus arbitration with master devices on the CAN bus that are attempting to send the Battery Management System CAN command messages.



The time to transfer the total number of CAN report messages plus the inter-report interval per message must be less than the [CAN report interval](#) or there will be performance related issues with the CAN bus operation. Any additional CAN traffic on the bus from other devices must also be considered.

### 7.8.1.4. Command Message Mapping

The Battery Management System can be configured to accept specific CAN IDs as *command message* inputs to perform actions such as:

- Connect/disconnect a battery to the DC bus
- Clear faults and warnings
- Update the controller heartbeat

A CAN ID is configured as a command message by the configuration register settings:

```
sc_canbus_map[N].command = 1
sc_canbus_map[N].address = @<component[>.register>
```

Messages have a CAN IDs of `sc_canbus.base_can_address + N`.

This command configuration means that if a CAN message with the correct CAN ID is transmitted on the CAN bus, the BMS will receive the message and write the data received into the configured register address. The data sent with the message must match the expected size and sign of the register.

**Table 24. CAN Command Register Configuration**

Register	Operation
stack_trigger_summary[0].clear_faults	Set to 1 to clear faults
stack_trigger_summary[0].clear_warnings	Set to 1 to clear warnings
stack_control[0].requested_state	Set to 1 or 0 to connect or disconnect (respectively) a battery from the DC bus
sc_controller_heartbeat[0].value	Write any value to update the controller heartbeat and prevent the watchdog timer from expiring (if configured)

## 7.8.1.5. Special Application Note: Conditional Message Transmission

In some cases, it may be desirable to only broadcast a message when a certain condition is true. For example, an application may require that a message be broadcast when a GPI is set. This can be achieved by utilizing bulk reporting and setting `sc_canbus_bulk_map[0].baseenabledaddress` to the same thing as `sc_canbus_bulk_map[0].baseaddress`. Thus, the following configuration would result in Nuvation Energy BMS only transmitting a message when `GPI[0]` is 1:

**Table 25. Special Application of Bulk CAN Reporting**

Register	Setting
sc_gpi[0].address_enabled	1
sc_gpi[0].inverted	0
sc_canbus_bulk_map[0].baseaddress	@sc_gpi[0].value
sc_canbus_bulk_map[0].baseenabled	@sc_gpi[0].value
sc_canbus_bulk_map[0].numtoread	1

## 7.8.2. RS485 Modbus RTU

Refer to [Section 8.1, "Modbus Protocol Support"](#) for more information on using this interface.

The slave device address used by the BMS for Modbus RTU may be customized as required.

`sc_modbus_rtu.device_address`

- Set to the desired Modbus RTU slave device address

## 7.9. Measurement Calibration

The Nuvation Energy BMS provide calibration settings which allow the system to be fine-tuned for integration with a variety of measurement sensors. The preset values that ship with the BMS can be adjusted as required.

### 7.9.1. Thermistor Calibration

Nuvation Energy BMS can be configured to use any thermistor. A function that converts measured voltage into temperature must be determined and configured for the particular thermistor in use.

A sixth-order polynomial is used within the BMS to model this transfer function:

$$T(v) = C_0 + C_1 \cdot v + C_2 \cdot v^2 + C_3 \cdot v^3 + C_4 \cdot v^4 + C_5 \cdot v^5 + C_6 \cdot v^6$$

`stack_therm_poly.coeff0` (Factory)

- Set to  $C_0$  (Floating-point value)

`stack_therm_poly.coeff1` (Factory)

- Set to  $C_1$  (Floating-point value)

`stack_therm_poly.coeff2` (Factory)

- Set to  $C_2$  (Floating-point value)

`stack_therm_poly.coeff3` (Factory)

- Set to  $C_3$  (Floating-point value)

`stack_therm_poly.coeff4` (Factory)

- Set to  $C_4$  (Floating-point value)

`stack_therm_poly.coeff5` (Factory)

- Set to  $C_5$  (Floating-point value)

`stack_therm_poly.coeff6` (Factory)

- Set to  $C_6$  (Floating-point value)

The thermistor voltage is read by a 10 k $\Omega$  pull-up to 3.00 V DC. The first step in calculating coefficients for a thermistor is to create a table in Microsoft Excel™ or equivalent spreadsheet application with the following columns:

Temperature ( °C )	Resistance ( $\Omega$ )	V <sub>adc</sub> ( V )
-40	334274	2.91286
-35	241323	2.88063
....	....	....
125	336.75	0.09773

Temperature and resistance values are taken from the datasheet of the thermistor.  $V_{adc}$  is calculated using the following formula:

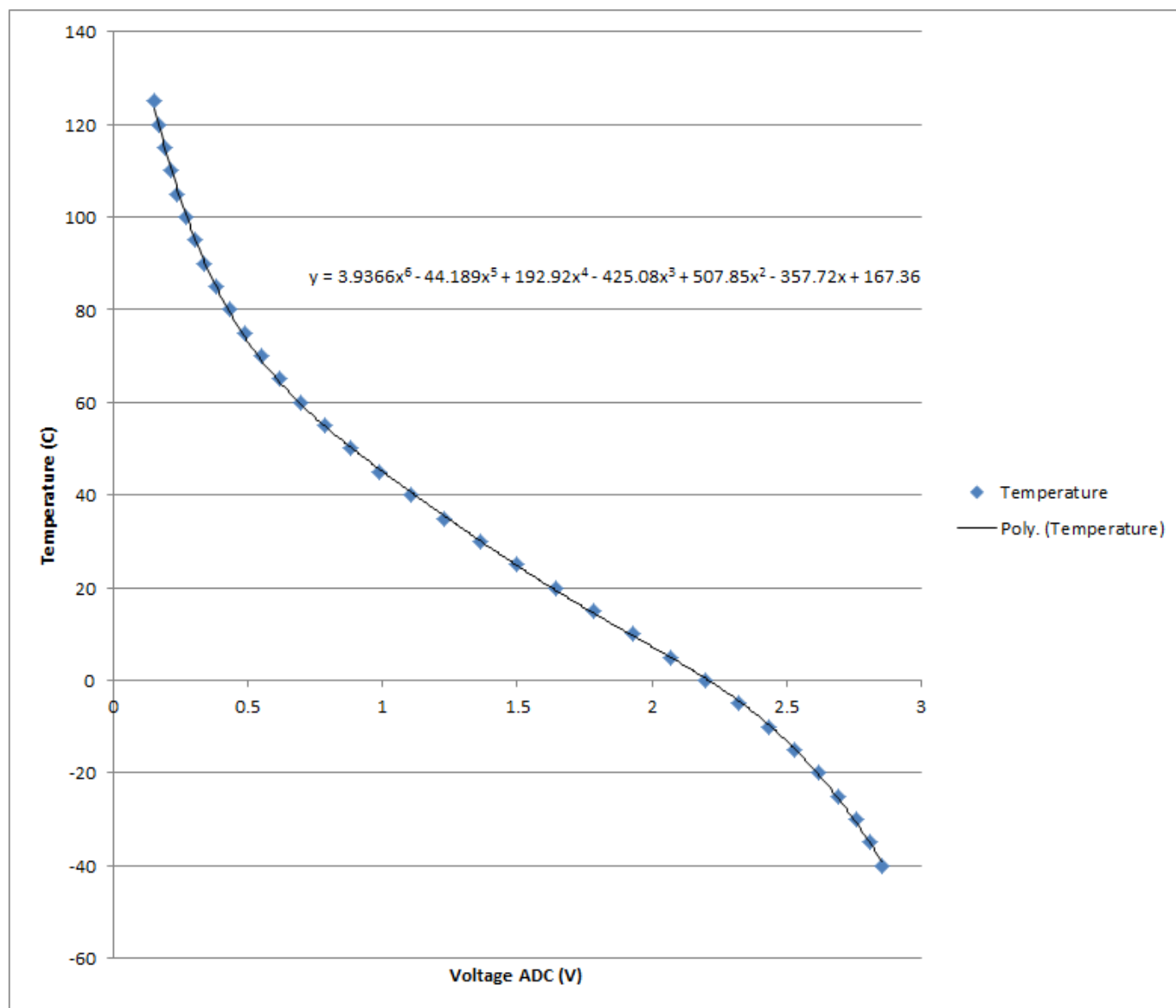
$$V_{adc} = 3.0 \left( \frac{R}{R + 10000} \right)$$

Using the line plot feature, create a graph of  $V_{adc}$  vs Temperature and turn on the trend line. Then modify the trend line to be a 6th order polynomial type, and display the equation on the chart. The equation will look like:

$$T(V) = (151.68) + (-352.94)V + (549.33)V^2 + (-482.08)V^3 + (223.69)V^4 + (-51.518)V^5 + (4.5693)V^6$$

These polynomial coefficients can then be used to configure Nuvation Energy BMS for this thermistor.

An example plot of these measurements is shown in the next figure.



**Figure 77. Typical Voltage ADC vs Thermistor Plot**

## 7.9.2. Stack Current Calibration

Current measurements are made using a current sensing analog front-end (AFE) that may be configured for a wide range of current shunt resistances.

### 7.9.2.1. Stack Current Factory Calibration

The factory calibration of current readings is implemented according to the following formula that converts ADC readings into a current:



$$I(\text{current}_{adc}) = \text{current}_{adc} \left( \frac{\text{multiplier}}{\text{divider}} \right)$$

The multiplier and divider are set in configuration as follows.

`pi_afe_iadc.multiplier` (Factory)

- Calculated calibration setting
- Set as per current shunt selection

`pi_afe_iadc.divider` (Factory)

- Calculated calibration setting
- Set as per current shunt selection

Calculating an appropriate multiplier and divider is best illustrated through an example. Assume a 5 mΩ current shunt is chosen that will have 50 mV across its terminals at 10 A. This voltage across the shunt at 10 A will be used to determine the multiplier and divider.

The first step is to calculate the conversion factor between the measured voltage and the calculated current value using the following equation:

$$a = \frac{\text{current}}{\left( \frac{V_{adc}}{300} \right) 2^{31}}$$

In this example, the conversion factor is 0.000027939 (10000mA and  $V_{adc}$  is 50 mV). The next step is to express this factor according to the following equation:

$$a = \frac{\text{multiplier}}{\text{divider}}$$

The easiest approach is to use the divider to achieve the desired precision and then use the multiplier to achieve the desired accuracy, keeping in mind that both values must be expressed as integers. In this case, the divider can be 10000000 and the multiplier can then be 279.

Additional work can still be done to refine these values. Searching for more accurate multipliers and dividers through a spreadsheet or trial and error can reduce rounding error.

For example a divider of 111000 and a multiplier of 31 are more accurate by a decimal place than the previous values. Running experimental calibration using an external tool to measure current (such as a multimeter) provides the best estimate.



For older versions of firmware, specifically versions before 4.58.0, these settings were different. To migrate multiplier/divider settings from before 4.58.0 to later versions, ensure the multiplier is reduced by a factor of 256. For example: if the old

multiplier was 1024, the new multiplier should be  $1024/256 = 4$ .

## 7.9.2.2. Stack Current Field Calibration

Units that have been factory locked may still require tuning for variability in shunts on a per-unit basis. The equation for Stack Current Field Calibration adjustment is:

$$I(\text{rawcurrent}) = \text{rawcurrent} \left( 1 + \frac{\text{gain}}{100000} \right)$$

The gain is set in the configuration as follows.

`stack_power.current_gain` (Configuration)

- Fine tuning gain for raw current values.
- A value of 1000 = +1% adjustment ( $\text{rawcurrent} * 1.01$ )
- A value of -1000 = -1% adjustment ( $\text{rawcurrent} * 0.99$ ).

At the factory, appropriate limits for the gain are defined to ensure a user does not misconfigure the system. The gain limit is set in the configuration as follows:

`stack_power.current_gain_limit` (Factory)

- Absolute value of the limit for gains on raw current values.
- If the `stack_power.current_gain` register is set to outside of this limit, the gain applied will be limited to this value.

## 7.9.2.3. Charge Deadband Setting

In addition to calibrating current readings, it is also necessary to define the deadband that will be used to determine whether the battery stack is charging, discharging, or at rest (holding).

`stack_charge_status.hold_current` (Factory)

- The battery stack is considered at rest unless the absolute value of current exceeds this threshold
- Set as required by application (typically between 50 and 100 mA)

While this threshold is used to qualify certain aspects of SoC estimation, Coulomb counting takes place continuously regardless of the configured hold current value. Other functions that require knowledge of the charge or discharge state of the stack (e.g. the thermistor temperature thresholds) also make use of this threshold.

## 7.9.3. Stack Voltage Calibration

Stack voltage measurements are made using a voltage sensing analog front-end (AFE) that must be configured for Nuvation Energy BMS.

## 7.9.3.1. Stack Voltage Factory Calibration

Factory calibration operates according to the following formula that converts an ADC reading to a voltage:

$$I(voltageadc) = voltageadc \left( \frac{multiplier}{divider} \right)$$

`pi_afe_vadc.multiplier` (Factory)

- Default calibration: set to 1

`pi_afe_vadc.divider` (Factory)

- Default calibration: set to 1414

While these defaults are likely acceptable for most applications, the calibration can be fine-tuned as needed for particular situations.

## 7.9.3.2. Stack Voltage Field Calibration

Units that have been factory locked may still require tuning for variability on a per-unit basis. The equation for Stack Voltage Field Calibration adjustment is:

$$V(rawvoltage) = rawvoltage \left( 1 + \frac{gain}{100000} \right)$$

The gain is set in the configuration as follows.

`stack_power.voltage_gain` (Configuration)

- Fine tuning gain for raw voltage values. A value of 1000 = +1% adjustment ( $rawvoltage * 1.01$ ). A value of -1000 = -1% adjustment ( $rawvoltage * 0.99$ ).

At the factory, appropriate limits for the gain are defined to ensure a user does not misconfigure the system. The gain limit is set in the configuration as follows:

`stack_power.voltage_gain_limit` (Factory)

- Absolute value of the limit for gains on raw voltage values.
- If the `stack_power.voltage_gain` register is set to outside of this limit, the gain applied will be limited to this value.

## 7.10. Hardware Settings

For High-Voltage BMS systems, the following configuration registers must be set as per the Cell Interface variant in use for your application.

`sc_linkbus.softwareid`

- Set to 0 for NUV100-CI-12
- Set to 1 for NUV100-CI-16
- Set to 2 for NUV100-CI-4M12

`sc_linkbus.cicount`

- Set to the total number of Cell Interface modules connected to the Stack Controller

The power required by the Cell Interface may be sourced either from the battery cells themselves or from the 24 V DC supply of the BMS through the *Link Bus*.

`sc_linkbus.power_mode`

- Set to 0 to power Cell Interface modules from battery cell power
- Set to 1 to power Cell Interface modules from the Stack Controller through the *Link Bus*

## 7.10.1. Link Bus Scan Period

The rate at which cell voltage and temperature measurements are updated is determined by a configurable *Link Bus* scan period. All cell voltages are measured every scan cycle. Thermistor temperature measurement is multiplexed across eight consecutive scan cycles. One thermistor in each group of cells (monitored by a single BMS module) is measured during each cycle, so the effective scan period of any given thermistor in the system is 8 times the *Link Bus* scan period.

`sc_linkbus.scan_period`

- Measurement period for cell voltages
- This is typically set to 1 second

Performance of passive balancing is also closely related to the scan period. In Nuvation Energy BMS, a single wiring harness is used to measure cell voltage as well as bleed off passive cell balancing current. It is not possible to make accurate voltage measurements while simultaneously balancing cells. To ensure accurate measurement, balancing current flow must stop before measurement can start (and any transient effects associated with that current flow must be allowed to settle). This is handled automatically by Nuvation Energy BMS. This means that passive balancing has some duty cycle that is less than 100% in practical systems.

The *Link Bus* has a configurable settling period for balancing that corresponds to the amount of time balancing is left off before voltage measurements are taken.

`sc_linkbus.balance_settle_period`

- Settling period during which balancing is disabled before cell voltage measurement.
- Typically set between 50 and 100 ms

For most systems, values between 50 and 100 ms yield accurate, stable measurements. When coupled with a scan period of 1000 ms, balancing duty cycles above 90% are achievable.

For some applications, it may be desirable to de-rate the effective passive balancing current by increasing the balancing settling period as a way to decrease the balancing duty cycle. If the balancing settling period is increased beyond the scan period, the actual scan rate of the system will start to decrease below the configured value. If this is increased even further, the reduced scan period will eventually trip the scan rate watchdog, putting the system into the fault state.

## 7.10.2. Cell Voltage Low-Pass Filter

One of two low-pass filters (LPF) are available for cell voltage measurement. This cell voltage low-pass filter is also referred to as the 'Measurement Anti-Aliasing Filter'. The filters are implemented in hardware and are selected via the following configuration.

`sc_linkbus.cell_voltage_lpf_bw`

- A value of 0 selects a 7 kHz LPF (i.e OFF)
- A value of 1 selects a 26 Hz LPF (i.e. ON)



The 26 Hz LPF results in slower cell voltage scan rates. Refer to the measurement rates listed in [Maximum Stack Deployment](#).



The 26 Hz LPF cannot be used with the Cell Interface - 12 V 4 channel

## 7.10.3. Cell Open Wire Detection and Diagnostics

The Battery Management System is capable of detecting wiring and circuit problems on each cell voltage channel. The following scenarios are detectable:

- Open circuits
- Short circuits
- High impedance connections/wiring
- Internal Battery Management System hardware failures

All channels are scanned for wiring and circuit problems when exiting lockout and during Battery Management System power up. Open wires and circuit problems are reported for each cell through the `cell.open_wire` flag. The common configurable settings for open wire detection on all modules are listed and described below.

`sc_linkbus.cell_open_wire_disabled`

- Typically this is set to 0
- Setting this to 1 will disable all open wire reporting

`sc_linkbus.cell_open_wire_period`

- The configurable period for scanning all channels for open wires
- Typically this is set to 3600000000 (one hour)



Open wires are only scanned according to this period when the batteries are

connected to the DC bus during typical operation.

#### `sc_linkbus.cell_open_wire_thresh`

- The configurable count for positive open wire detections before the `cell.open_wire` flag is set
- This is to ensure noise triggered false positives are filtered
- Typically this is set to 3

#### `sc_linkbus.cell_open_wire_max`

- The configurable maximum count for positive open wire detections
- This variable is used to prevent windup of each channel's counter
- It also dictates how many *Link Bus* scan cycles should be used to collect enough samples to detect open wires
- Typically set to 10 and cannot be less than `sc_linkbus.cell_open_wire_thresh`

For systems using Cell Interface - 12 channel or Cell Interface - 16 channel modules, the configurable settings are listed and described below.

#### `sc_linkbus.cell_open_wire_ratio_lower`

- The ratio below which the channel has an error, represented as an integer between 0 and 1000, (0 = 0.0, and 1000 = 1.0)
- Channels that have low ratios are indicative of high impedance connections between the cell and channel input, possibly due to loose connections or bad terminals.
- Very low ratios indicate an open wire or a Battery Management System hardware failure
- Typically set to 800, representing 0.8

#### `sc_linkbus.cell_open_wire_ratio_upper`

- The ratio above which the channel has an error, represented as an integer between 0 and 1000, (0 = 0.0, and 1000 = 1.0)
- Channels that have high ratios are indicative of either short circuits or a Battery Management System hardware failure
- This is typically 970, representing 0.97

For High-Voltage BMS systems using Cell Interface - 12 V 4 channel modules, the configurable settings are listed and described below.

#### `sc_linkbus.cell_open_wire_bal_tol`

- The maximum voltage spread within a CI4M12 cell measurement
- A high voltage spread within a CI4M12 channel indicates an open wire
- This is typically set to 500 for 500 mV

#### `sc_linkbus.cell_open_wire_low_thresh`

- Voltage below which a CI4M12 cell is considered to have an error
- Low voltages can indicate either an open or short circuit

- This is typically 100 for 100 mV

## 7.10.4. Fault Pilot Signal

The *Fault Pilot* signal is signal internal to the Nuvation Energy BMS used to open the contactors through a secondary control path in the case of a fault condition. When the system is faulted, the *Fault Pilot* signal should be suppressed to guarantee the contactor coils are de-energized, regardless of the state of the coil control software.

In most applications, this signal is a delayed version of the overall fault state of the BMS. This small delay allows for the opening of directional contactors according to any delays configured as part of the contactor output settings ([Section 7.7.1, "Contactor Outputs"](#)).

In Nuvation Energy High-Voltage BMS systems, the fault pilot is a signal generated by the Stack Controller and sent to the Power Interface over the *Stack Bus* ([Section 9.1.6, "Stack Bus"](#) and [Section 9.2.4, "Stack Bus"](#)).

`sc_gpo_fault_pilot.fault_enabled`

- Set to 1 to enable Fault Pilot signal to assert when `stack_factory_trigger_summary.fault` is tripped

`sc_gpo_fault_pilot.fault_delay`

- Set according to application (typically, between 100 and 5000 ms)

## 7.10.5. Under-Voltage Shutdown

It may be possible to implement an under-voltage shutdown in the High-Voltage BMS depending on your application. This may involve setting `stack_uvlo_cell_voltage.thresh` and `stack_uvlo_stack_voltage.thresh`. Please contact [support@nuvationenergy.com](mailto:support@nuvationenergy.com) for more information.

## 7.10.6. Clock Verification

The Stack Controller and Power Interface module each have an independent clock and these are checked against each other regularly to ensure proper operation. The following components are associated with clock checking diagnostics.

`sc_fault_clocks.disabled`

- Setting to 1 enables clock checking on the Stack Controller
- Disabling this fault will prevent the BMS from tripping a fault if the Stack Controller detects that the Power Interface clock is out of sync
- Typically set to 1

`pi_clock_check.enabled`

- Setting to 1 enables clock checking on the Power Interface
- No fault is associated with this feature and the Power Interface will directly disconnect the contactors if a clock error is detected

- Typically set to 0

## 7.10.7. Open Shunt Detection and Diagnostics

Nuvation Energy BMS is capable of detecting improperly installed, loose, or disconnected sense wires that are used to measure the stack voltage and stack current by tracking the correlation between the two measurements.

`stack_open_shunt_detector.enabled`

- Set to 1 to enable the open shunt detection algorithm

`stack_open_shunt_detector.current_activity_threshold`

- Current measurements above this threshold are used for tracking the correlation between stack voltage and stack current
- Typically set to the same value as the `stack_short_shunt_detector.current_activity_threshold` register

`stack_open_shunt_detector.minimum_resistance_threshold`

- The threshold value below which resistance estimates are considered faulty
- The threshold value ( $R_T$ ), calculated in  $\Omega$ , can be determined using the equation below, where ( $R_c$ ) is the cell resistance in  $\Omega$  and ( $N_{cell}$ ) is the number of cells connected in series for a stack
- The value of the BMS register `minimum_resistance_threshold` in  $\mu\Omega$  can be calculated by multiplying the value of  $R_T$  by 1000000

$$R_T = \frac{R_c \cdot N_{cell}}{10000}$$

`stack_open_shunt_detector.filter_sample_count`

- This register is used to adjust how much noise is filtered out to prevent false trips due to noisy signals
- A higher value will result in a greater amount of filtering
- Typically set to 10

`stack_fault_open_shunt.thresh`

- The minimum percent of faulty samples required to trip an open shunt fault
- These faulty samples are current and voltage measurements that do not correlate with each other
- Typically set to 50

## 7.10.8. Short Shunt Detection and Diagnostics

Nuvation Energy BMS can detect shorts in the stack current and stack voltage sensing circuit by tracking the correlation between the stack voltage and stack current.



## `stack_short_shunt_detector.enabled`

- Set to 1 to enable the short shunt detection algorithm

## `stack_short_shunt_detector.current_activity_threshold`

- If a short circuit exists, the measured current would fall below this threshold value
- The threshold value ( $I_a$ ), calculated in A, can be determined using the equation below, where ( $I_{fuse}$ ) is the fuse rating of the stack in A
- The value of the BMS register `current_activity_threshold` in mA can be calculated by multiplying the value of  $I_a$  by 1000

$$I_a = 0.0056 \cdot I_{fuse}$$

## `stack_short_shunt_detector.voltage_activity_threshold`

- If voltage activity is detected above this threshold value, the detector looks for correlation between the stack voltage and stack current.
- The threshold value ( $V_a$ ), calculated in V, can be determined using the equation below, where  $I_a$  is current activity threshold value calculated above and  $R_{stack}$  is the stack resistance in  $\Omega$
- The value of the BMS register `voltage_activity_threshold` in mV can be calculated by multiplying the value of  $V_a$  by 1000

$$V_a = 10 \cdot I_a \cdot R_{stack}$$

If the value of the stack resistance is not known, it can be calculated using the cell resistance ( $R_c$ ) in  $\Omega$  and the number of cells in the stack ( $N_{cell}$ ).

$$R_{stack} = N_{cell} \cdot R_c$$

## `stack_short_shunt_detector.current_time_constant`

- The time constant of the filter used to remove noise in the current signal
- Typically set to 1000000

## `stack_short_shunt_detector.voltage_time_constant`

- The time constant of the filter used to remove noise in the voltage signal
- Typically this is set to 1000000

## `stack_short_shunt_detector.activity_detection_time`

- The settling time of the filter used to detect gradual changes in stack voltage The value also represents the time required to detect these changes.
- Typically set to 1800000000

## `stack_short_shunt_detector.ramp_detection_time`

- The time constant of the filter used to detect rapid changes in stack voltage. The value also represents the time required to detect these changes.
- Typically set to 10000000

## `stack_short_shunt_detector.filter_sample_count`

- Used to adjust how much noise is filtered out to prevent false trips due to noisy signals
- A higher value will result in a greater amount of filtering
- Typically set to 100

## `stack_fault_short_shunt.thresh`

- The minimum percent of faulty samples required to trip a short shunt fault
- These faulty samples are current and voltage measurements that do not correlate with each other
- Typically this is set to 50



If the short shunt fault has tripped, it cannot be cleared by pressing Clear Faults until the activity falls below its threshold. It may take several minutes for this transition to occur. Refer to [Section 10.1.7.1.2, "Clearing the Short Shunt Fault Is Not Immediate"](#) on how to work around this issue.

## 7.10.9. Temperature Consistency Check

For each Cell Interface module in the stack, the temperature consistency check will calculate the maximum deviation between the smallest and largest measured temperature.

If the deviation exceeds a configurable threshold value, a fault will be tripped.

### `stack_fault_ci_therm_consistency.thresh`

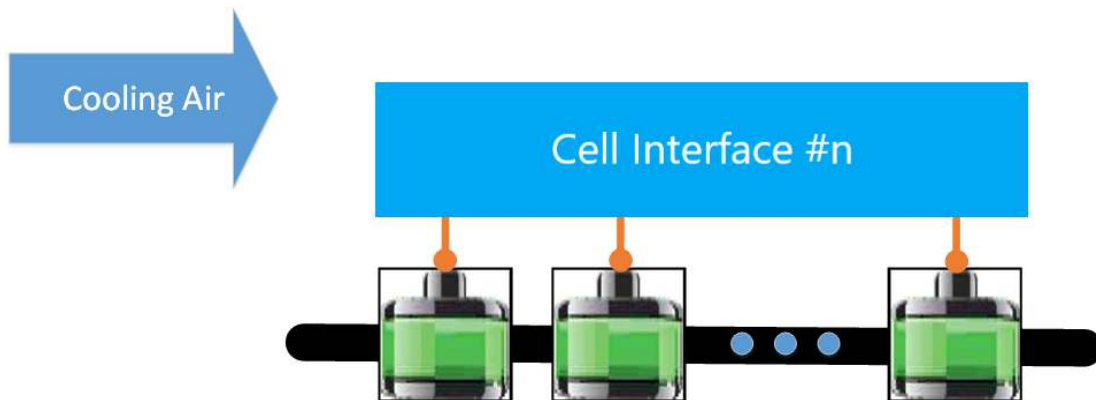
- The maximum temperature deviation, in units of °C, that is allowed between thermistors of a single Cell Interface module
- Typically this is set to 15

In order to minimize this temperature deviation and prevent false trips, consider the following recommendations below.

### 7.10.9.1. Recommendation for Temperature Consistency Check

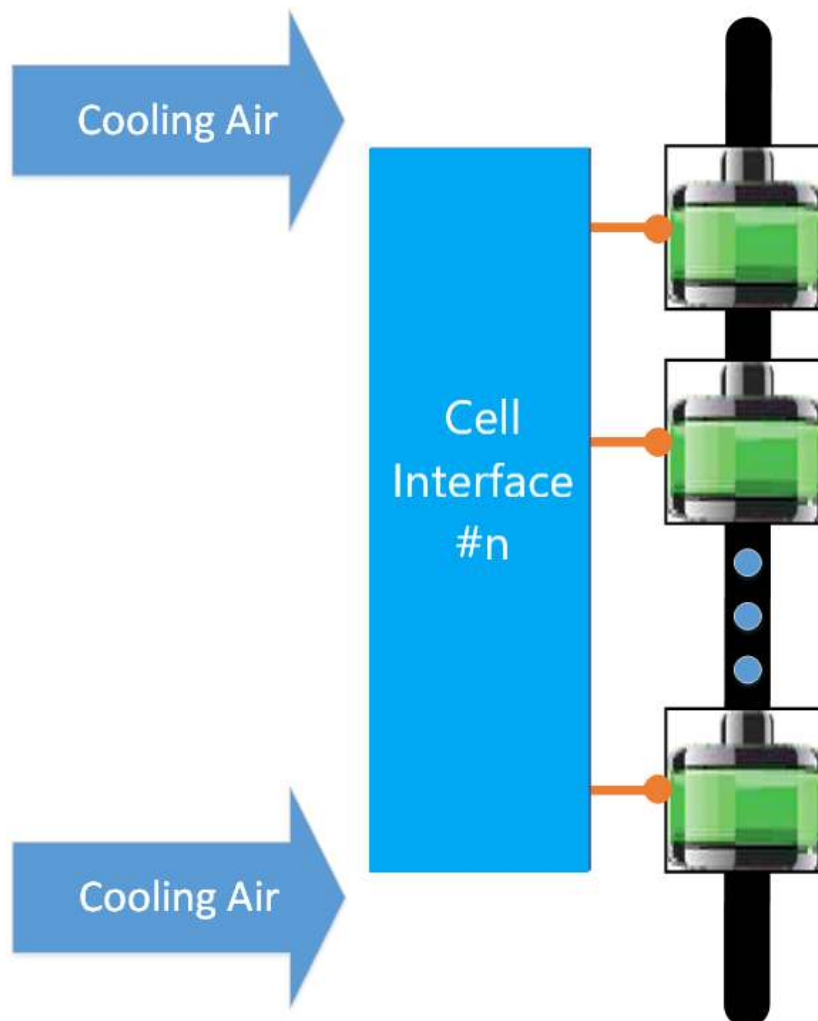
There are two main factors that need to be considered when connecting the thermistors. First, it is important to ensure that the thermal cooling environment for each of thermistors is similar. One approach is to place the thermistors orthogonal to the cooling air direction.

Consider the following scenario where the cooling air flow is in the same direction as the installed thermistors.



**Figure 78. Thermistor arrangement scenario one**

A temperature gradient will develop in the direction of the air flow, where the cells to the far left will be cooled more than the ones to the right. This can result in a large temperature deviation within the cell grouping. To minimize this, the cells can be rearranged as shown in below.



**Figure 79. Thermistor arrangement scenario two**

The effect of the temperature gradient can be attenuated by grouping thermistors orthogonal to the gradient. During operation, the temperature measurements within this group increase and decrease and therefore, when a failure occurs, the problematic thermistor will begin to deviate from the rest of the group.

A second factor that can be considered to minimize the temperature deviation within cell grouping is battery State-of-Health. It is advantageous to group cells with similar State-of-Health values together since they would have similar thermal responses to current activity.

## 7.10.10. Contactor Life Tracking

Nuvation Energy BMS provides a mechanism to track the life of each contactor and to trigger a fault when it is end-of-life. The life of each contactor at any given time is determined by the various opening currents the contactor has experienced. Each contactor can be individually configured for contactor life tracking where `ssg_contactor_monitor[n]` and `ssg_contactor_life_tracker[n]` corresponds to `stack_contactor[n]`.

Typically, a contactor will specify the number of make/break cycles it is rated for within a current range. For example, if contactor XYZ has the following ratings:

- 1000V, 200A max, 50 cycles make/break
- 1000V, 100A max, 5000 cycles make/break

Depending on the application, it may be safe to assume that above a number of rated make/break cycles, contactor life tracking is not required. In this example, we will assume that anything above 5000 cycles does not need to be tracked.

This will translate to the following contactor life tracking configuration:

- `ssg_contactor_monitor[0].enabled = 1`
- `ssg_contactor_life_tracker[0].enabled = 1`
- `ssg_contactor_life_tracker[0].min_opening_current = 100`
- `ssg_contactor_life_tracker[0].max_opening_current = 200`
- `ssg_contactor_life_tracker[0].maxRatedCurrent = 199`
- `ssg_contactor_life_tracker[0].maxLife = 50`

A complete list of all contactor life tracking registers is shown below.

`ssg_contactor_monitor[n].enabled`

- Set to 1 to enable monitoring of contactor `n` for contactor life tracking

`ssg_contactor_life_tracker[n].enabled`

- Set to 1 to enable contactor life tracking for contactor `n`

`ssg_contactor_life_tracker[n].is_precharge`

- Set to 1 if the contactor is configured as the precharge contactor

`ssg_contactor_life_tracker[n].current_capture_window`

- Window of current samples to consider for an opening current, centered around when the contactor opens. The largest value in this window is used
- Typical value is 2000000 (2 seconds). This would mean current measurements taken 1 second before and 1 second after the contactor opening event will be considered

`ssg_contactor_life_tracker[n].min_opening_current`

- Minimum opening current, above which the contactor life will increment
- This is typically set to the minimum current at which the contactor make/break cycles is limited

`ssg_contactor_life_tracker[n].max_opening_current`

- Maximum opening current, above which the contactor is immediately considered end-of-life

`ssg_contactor_life_tracker[n].maxRatedCurrent`

- Current to assume the contactor opened at if the opening event could not be monitored
- Typically set to between `ssg_contactor_life_tracker[n].min_opening_current` and `ssg_contactor_life_tracker[n].max_opening_current`

`ssg_contactor_life_tracker[n].maxLife`

- Maximum life of the contactor, above which the contactor is considered end-of-life
- This should be set to the number of make/break cycles the contactor is rated for between `ssg_contactor_life_tracker[n].min_opening_current` and `ssg_contactor_life_tracker[n].max_opening_current`

`ssg_warn_contactor_end_of_life.thresh`

- The threshold below which a warning is triggered around the contactor's remaining life
- This value is a Percentage, indicating the remaining life of the contactor

`ssg_fault_contactor_end_of_life.thresh`

- The threshold below which a fault is triggered around the contactor's remaining life
- This value is a Percentage, indicating the remaining life of the contactor. This should be set to 0

Please contact [support@nuvationenergy.com](mailto:support@nuvationenergy.com) if this feature is required for your application's functional safety.

## 7.11. Watchdogs and Freshness

Watchdog timers are employed in Nuvation Energy BMS to ensure specific functions execute regularly. If a watchdog timer is not periodically reset, the timer will expire and an associated fault will trip and disconnect the contactors. The watchdog timers are exposed by the BMS through multiple watchdog components.

`wdt.expired`

- 1 indicates that the timer has not been reset within the configured time and is considered expired
- 0 indicates the timer has not expired.

`wdt.period`

- The time period within which the timer needs to be reset before it expires

The watchdogs configurations are described below.

**Table 26. Nuvation Energy BMS Watchdog Components**

Watchdog Component	Notes
<code>sc_ram_test_wdt</code>	Configures the watchdog timer for the RAM testing. Typically period is 60000000 (60 s).
<code>sc_rom_test_wdt</code>	Configures the watchdog timer for the ROM testing. Typically period is 60000000 (60 s).

## 7.11.1. Freshness

Freshness is a mechanism in Nuvation Energy BMS to guarantee data within the system is up to date. Measurements such as cell voltage, cell temperatures, and stack voltage are captured on the Cell Interface modules and the Power Interface and are aggregated on the Stack Controller. Points of failures on the Cell Interface modules or Power Interface, or on the Link Bus or Stack Bus can result in data reported by the BMS to be stale. Regardless of how and where data stops being propagated, freshness provides a mechanism for the system to detect these failures and perform the necessary procedures to keep the system safe.

The freshness mechanism is exposed by the BMS through multiple freshness component.

`fresh.stale`

- 1 indicates that the associated data has not been updated within the configured time and is considered stale
- 0 indicates the data is still fresh

`fresh.period`

- The time period within which the associated data needs to be updated before it is considered to be stale

Freshness components are mapped to certain types of data such as cell data, stack data, and so on. When a freshness component becomes stale, it means that its associated data has not been updated within the configured time period.

**Table 27. Nuvation Energy BMS Freshness Components**

Freshness Component	Associated Data	Notes
<code>stack_cell_fresh</code>	<code>cell.voltage</code> <code>`cell.open_wire`</code>	Ensures cell information is updated. Only installed cells are checked for freshness. The period must be longer than <code>sc_linkbus.scan_period</code> , typically 3000000 (3 seconds).

Freshness Component	Associated Data	Notes
stack_therm_fresh	therm.temperature `therm.circuit_fail`	Ensures thermistor information is updated. Only installed thermistors are checked for freshness. The period must be longer than 8x sc_linkbus.scan_period, typically 10000000 (10 seconds).
sc_ci_fresh	ci.supply_error ci.mux_error	Ensures Cell Interface information is updated. Only installed Cell Interface modules are checked for freshness. The period must be longer than sc_linkbus.scan_period, typically 3000000 (3 seconds).
stack_cell_open_wire_fresh	stack_cell_summary_stat.cell_open_wir e_executions	Ensures open-wire detection occurs periodically. The period must be set longer than sc_linkbus.cell_open_wire_period, typically 12000000 (120 seconds).
stack_power_fresh	stack_power.voltage `stack_power.current`	Ensures stack power information is updated. The period must be set longer than pi_afe.sample_period, typically 60000000 (60 seconds).
stack_linkbus_fresh	sc_linkbus.power_error	Ensures Link Bus checks occur periodically. The period must be longer than sc_linkbus.scan_period, typically 3000000 (3 seconds).
sc_factory_verify_fresh	sc_factory_persist.verification_error	Ensures Stack Controller factory configuration verification occurs periodically. The period is typically 60000000 (60 seconds).
pi_factory_verify_fresh	pi_fault_config_factory_verify.error	Ensures Power Interface factory configuration verification occurs periodically. The period is typically 60000000 (60 seconds).
ssg_contactor_life_verify_fresh	ssg_contactor_life_persist.verificati on_error	Ensures verification of the contactor life data occurs periodically. The period is typically 60000000 (60 seconds)
ssg_contactor_end_of_life_fresh	ssg_contactor_life_tracker_summary.mi n_life_remaining	Ensures contactor life is evaluated periodically. The period is typically 60000000 (60 seconds)
ssg_contactor_monitor_fresh	ssg_contactor_monitor.status	Ensures contactor monitoring occurs periodically. The period is typically 10000000 (10 seconds)

## 8. Communication Protocols

### 8.1. Modbus Protocol Support

Nuvation Energy BMS implements the SunSpec battery models defined in the Modular Energy Storage Architecture (MESA) as the top-level Modbus interface to the product. Specifically the BMS implements the MESA Draft 3 Storage models (800 Series).

The specifications for these models are available for download at <https://www.nuvationenergy.com/technical-resources>.



MESA Draft 3 is currently implemented on Nuvation Energy BMS.

#### 8.1.1. Modbus RTU

This protocol is used in serial communications. The default configuration is as follows:

- Baud rate: 38400
- Parity: even
- Data bits: 8
- Stop bits: 1

The Modbus RTU slave address must be set through software configuration. The default slave address value is 1.



A slave address of zero is used as a broadcast address and should not be used as the Modbus RTU slave address.

#### 8.1.2. Modbus TCP

This protocol is used for communications over TCP/IP networks. A single Modbus TCP connection is supported over port 502 for read and write access. All register data is transmitted as big endian (most significant byte first).

#### 8.1.3. Implemented MESA Models

The MESA standards contain a number of 'models' that can be implemented by vendors to describe a storage device at various levels of detail. The models implemented by Nuvation Energy BMS are described in the sections below.

Detailed register maps for all Draft 3 models are found in the document *MESA-Energy-Storage-Information-Models\_Draft\_3.xlsx* found at the Nuvation Energy technical resources located at <https://www.nuvationenergy.com/technical-resources>.

##### 8.1.3.1. Common Model



This model primarily contains information to identify the device (e.g. manufacturer, model, serial number) as well as the version of software running on the device. The Modbus address of this model is 40000. [Table 28, "Common Model Points"](#) describes the points of the Common Model as implemented in the BMS. A full description of the Common Model can be found in the SunSpec specification bundle.

**Table 28. Common Model Points**

Point Name	Address	Description	Note
SunSpec_ID	40000	Identifies this as a SunSpec Modbus Map	Set to 0x53756e53
ID	40002	Identifies Common Model Block	Set to 0x0001
L	40003	Length of block	Set to 66
Mn	40004	Device Manufacturer	Set to "Nuvation Engineering"
Md	40020	Device Model Number	Contents of BMS registers sc_mfg_info.manufacturing<0-3>
Opt	40036	Options	Set to "Fully Loaded"
Vr	40044	Version Information	Contents of the sc_fw_image.build_version and sc_fw_image.build_id BMS registers separated by a colon
SN	40052	Device Serial Number	Contents of BMS registers sc_mfg_info.serialnumber<0-1>
DA	40068	Device Address	Set to 0x1234

### 8.1.3.2. S801

This model describes an energy storage device at the highest possible level. State-of-Charge and overall alarm and warning states are found here. All mandatory points are implemented. The Modbus address of this model is 40070.

### 8.1.3.3. S802

This model describes a battery storage device. At this level, the critical operational information includes the charge and discharge current limits. All mandatory points are implemented. The Modbus address of this model is 40094.

### 8.1.3.4. S803

This model describes a lithium-ion battery in detail. Voltage, temperature, and current statistics are available at the pack and stack level within this model. All mandatory and most optional points are implemented. The Modbus address of this model is 40116.

### 8.1.3.5. End Model

This model marks the end of the implemented Modbus address space.

## 8.1.4. MESA Model Structure and Nomenclature

This section is a clarification of terms used to describe a MESA model. It is used to understand the terminology in the spreadsheet presented in the previous section.

### 8.1.4.1. Points

All MESA models are a collection of points (i.e. Modbus registers). These points can be one or more Modbus registers in length. By definition, each Modbus register is 16 bits wide. For points that are larger than one Modbus register, partial read accesses are not allowed. A modbus request to a MESA point must read all registers that make up the point. Otherwise, a Modbus read/write error is returned on such an access.

For example, the MESA s802 model has the Evt1 point which is of type bitfield32. In this case, the point spans 2 Modbus register and so a request to read this point must read 2 modbus registers.

### 8.1.4.2. Fixed/Repeating Blocks

MESA models are described as collections of Fixed and Repeating blocks of points. A Fixed block is a set of points that is always defined and never changes in its size. A Repeating block describes a set of related points (i.e. usually for a string of batteries) of which there could be multiple instances of the Repeating blocks. The points within a repeating block are the same but these sets of blocks are concatenated sequentially.

For example in the 803 model, there is a set of repeating blocks that describe data for a particular stack/string of batteries. Accessing the 803 repeating block corresponds to using a stack/string index (0, 1, 2, ...) to access the desired repeating block.

For a single-stack Nuvation Energy BMS there is only one 803 repeating block. If a stack/string is configured in Nuvation Energy BMS software to be installed, then accessing its 803 repeating block will provide a valid Modbus response.

Repeating blocks are taken into account in the length indicated in the model header.

### 8.1.4.3. Unimplemented Points

Any MESA point that is not implemented by a vendor will generate an unimplemented response. The response will be a valid Modbus read response but all point data returned will report unimplemented values. The unimplemented values vary by type as listed in the following table.

**Table 29. Unimplemented Point Values**

Type	Width (bits)	Unimplemented Value (hexadecimal)
signed int	16	0x8000
unsigned int	16	0xFFFF
signed int	32	0x80000000
unsigned int	32	0xFFFFFFFF
enumeration	16	0xFFFF

Type	Width (bits)	Unimplemented Value (hexadecimal)
enumeration	32	0xFFFFFFFF

A write to a writeable MESA point that is unimplemented will generate a Modbus write error.

The MESA implementation on Nuvation Energy BMS has unimplemented points as follows:

**Table 30. Unimplemented MESA Points**

Model	Point Name	Description
801	Evt	Event status bit field
801	DisChaRate	Self Discharge Rate
801	MaxRsvPct	Maximum Reserve Percent
801	MinRsvPct	Minimum Reserve Percent
801	ChaSt	Charge Status
801	DerAlarmReset	Alarm Reset (read is unimplemented)
802	CycleCt	Cycle Count
802	WMaxDisChaRte	NamePlate Max Discharge Rate
802	SoH	State of Health (reported in 803 model per string)
802	BatReqPCSSt	PCS State Request
802	BatReqW	Battery Power Request
802	SoH_SF	State of Health Scale Factor
802	BatReqW_SF	Battery Power Request Scale Factor
803	StrEvt2	String Event 2
803	StrConFail	String Connection Fail

#### 8.1.4.4. Scale Factors

All MESA points are integer values (signed or unsigned). To account for different range values beyond the data size (i.e. greater than 65535 for an unsigned 16 bit value) or some fractional value (i.e. 1.1), some MESA points have scale factors associated with them. The scale factor is another point within the model which contains a signed integer exponent of base 10 that scales a corresponding point value. For example, a scale factor of 2 would result in multiplying the corresponding point by 100. Likewise a scale factor of -3 would result in a scale factor of 0.001. Refer to the prior MESA Draft 3 spreadsheet for the relationship between MESA points and their scale factors. All scale factors are fixed for a model and do not change in value.

#### 8.1.5. Operational Cases for MESA

There are two main operational cases for the control of Nuvation Energy BMS over its MESA interface:

1. An external controller (sometimes called "Energy Storage Controller") is used to coordinate power control functions of the BMS in conjunction with some other equipment (such as an inverter). This controller requires periodic and rapid responses of MESA point reads as well as some control over

the operation of the BMS (such as stack connectivity). If there is a loss of communication between this controller and the BMS, the BMS will disconnect the stack(s) as a safety precaution.

2. An owner/operator of a battery system requires control of the BMS to monitor the activity of the batteries and track battery usage and its charge/discharge activities. This information can then be used to characterize the usage of the Battery Management System and to validate battery warranties of a vendor.

Read and write Modbus TCP operations can be performed over the standard Modbus port 502 (only a single connection is supported on this port).

The shortest polling periods expected for the MESA models during operation are listed below:

- All fixed model blocks: 100ms
- 801: less than 100ms
- 802: less than 100ms
- 803 Fixed Block: less than 100ms
- 803 Repeating Block(s): less than 100ms (single block), up to 1000ms (36 repeating blocks)

Depending on BMS system load and size, some requests could take up to 1000ms. The BMS typically samples data at between 250ms to 1000ms depending on configuration, so reading the data faster than this will not improve system behaviour and incurs unnecessary overhead.

These two operational cases will be discussed in detail in the following sections.

## 8.1.5.1. External Controller Communicating Over MESA Interface

An external controller typically polls Nuvation Energy BMS battery control points at a rate of 2–4 Hz. This controller reads data points required to manage current flow in the system. The following table summarizes the most important points an external controller may want to read from the BMS.

**Table 31. MESA Points Read by an External Controller**

Model	Block	Point Name	Address	Scale Factor	Purpose
801	Fixed	DERHb	40086	No	BMS Heartbeat counter incremented every second
802	Fixed	Evt1	40101	No	Bit field of all faults/warnings of a the BMS
802	Fixed	Vol	40105	Yes	External DC voltage of the battery system
802	Fixed	MaxBatACha	40106	Yes	Charge current limit calculated by the BMS
802	Fixed	MaxBatADischa	40107	Yes	Discharge current limit calculated by the BMS
803	Fixed	BTotDCCur	40127	Yes	Total DC current of the battery system

A controller may also want to command Nuvation Energy BMS to perform certain actions, such as connecting/disconnecting the battery. The following table provides the different writeable points in the MESA interface for different control functions:

**Table 32. MESA Points Written to by an External Controller**

Model	Block	Point Name	Address	Purpose
801	Fixed	ControllerHb	40087	Heartbeat register for external controller
801	Fixed	DERAlarmReset	40088	Clears all latched alarms in the BMS
802	Fixed	BSetOperation	40110	Commands Nuvation Energy BMS to connect/disconnect the battery

When configured, the heartbeat controller can be used to update the watchdog timer of Nuvation Energy BMS on single-stack and multi-stack implementations. If the heartbeat point is not updated within the watchdog timer period, a fault will be generated on the BMS stack and its corresponding contactors will be opened. The value written to the heartbeat point must increase in value and roll over to zero when the 16 bit range limit is reached.

The clearing of alarms of Nuvation Energy BMS is accomplished by writing a value of 1 to the DERAlarmReset point. This point is erroneously identified as read-only in the Mesa Draft 3 spreadsheet referenced earlier. Note that the alarm will not be cleared if the alarm condition is still present.



Only BMS Faults are cleared using this point. BMS Warnings are not cleared.

Commanding the Nuvation Energy BMS stack/string to connect/disconnect is accomplished through the BSetOperation point. This point accepts the enumerated values for these connect/disconnect operations. Additional connection logic (such as separate pre-charge circuitry) is managed automatically by the BMS when it is configured for such an operation.

## 8.1.5.2. External Nuvation Energy BMS Monitoring Over MESA Interface

An external data logger may want to access a variety of data from the BMS. In general, a data logger will not actively manage Nuvation Energy BMS; normally, it will not initiate actions such as connecting a battery stack to the DC voltage bus or clearing faults. A data logger should connect to one of the read-only Modbus connections (if available) to allow the writable Modbus connection to be available for separate external control functions. The following table contains the MESA data points exposed by the BMS that could be collected for logging purposes.

**Table 33. MESA Points Read by External Data Logger**

Model	Block	Point Name	Address	Scale Factor	Purpose
801	Fixed	SoC	40081	Yes	BMS State of Charge
801	Fixed	DERHb	40086	No	BMS Heartbeat counter incremented every second
802	Fixed	Evt1	40101	No	Bit field of all faults/warnings of a BMS
802	Fixed	Vol	40105	Yes	External DC voltage of the battery system
802	Fixed	MaxBatACha	40106	Yes	Charge current limit calculated by BMS
802	Fixed	MaxBatADischa	40107	Yes	Discharge current limit calculated by BMS
802	Fixed	BSetOperation	40110	No	BMS requested connection state of all stacks/strings

Model	Block	Point Name	Address	Scale Factor	Purpose
803	Fixed	BConStrCt	40118	No	Number of stacks/strings with contactor closed
803	Fixed	BMaxCellVol	40119	Yes	Maximum cell voltage measured
803	Fixed	BMaxCellVolLoc	40120	No	Module/String location of maximum cell voltage
803	Fixed	BMinCellVol	40121	Yes	Minimum cell voltage measured
803	Fixed	BMinCellVolLoc	40122	No	Module/String location of minimum cell voltage
803	Fixed	BMaxModTmp	40123	Yes	Maximum module temperature
803	Fixed	BMaxModTmpLoc	40124	No	Module/String location for maximum module temperature
803	Fixed	BMinModTmp	40125	Yes	Minimum module temperature
803	Fixed	BMinModTmpLoc	40126	No	Module/String location for minimum module temperature
803	Fixed	BTotDCCur	40127	Yes	Total DC current of the battery system
803	Fixed	BMaxStrCur	40128	Yes	Largest DC current reported by a stack/string
803	Fixed	BMinStrCur	40129	Yes	Smallest DC current reported by a stack/string
803	Repeat	StrSoC	40135 +Index	No	State of charge for a stack/string
803	Repeat	StrSoH	40136 +Index	Yes	State of health for a stack/string
803	Repeat	StrCur	40137 +Index	Yes	Current of a stack/string
803	Repeat	StrMaxCellVol	40138 +Index	Yes	Maximum cell voltage of a stack/string
803	Repeat	StrMinCellVol	40139 +Index	Yes	Minimum cell voltage of a stack/string
803	Repeat	StrCellVolLoc	40140 +Index	No	Location of min/max cell voltages of a stack/string
803	Repeat	StrMaxModTmp	40141 +Index	Yes	Maximum module temperature of a stack/string
803	Repeat	StrMinModTemp	40142 +Index	Yes	Minimum module temperature of a stack/string
803	Repeat	StrModTmpLoc	40143 +Index	No	Location of min/max module temperatures of a stack/string
803	Repeat	StrEvt1	40144 +Index	No	Alarms warnings and status bit field of a stack/string



The term Index in the Repeating block addresses used in the above table refers to a calculation of  $\text{Index} = \text{Stack Index} * \text{Length of Repeating block}$ . By definition, the 803 Repeating block is 16 Modbus registers in length.

## 8.1.6. Accessing MESA Models

MESA models are located contiguously in the Modbus address space starting at a base address of 40000. The Common Model is always located first in this space. The End Model is always last and is used to denote the end of MESA Modbus registers. Each model located between the Common Model and the End Model has a numeric identifier as well as a length. A handy tool that can be used to explore the MESA Modbus registers for Nuvation Energy BMS is modpoll.exe. It is available for free download at <http://www.modbusdriver.com/modpoll.html>.

Using modpoll.exe, the Common Model can be polled from a using the following command (assuming the device has an IP address of 192.168.1.21)

### Polling example with BMS IP address of 192.168.1.21

```
modpoll.exe -m tcp -0 -r 40000 -c 70 192.168.1.21

modpoll 3.4 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2013 proconX Pty Ltd
Visit http://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration...: address = 1, start reference = 40000 (PDU), count = 70
Communication.....: 192.168.1.21, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table

-- Polling slave... (Ctrl-C to stop)
[40000]: 21365
[40001]: 28243
[40002]: 1
[40003]: 66
.
.
.
[40068]: 4660
[40069]: -32768
```

As another example, the complete S802 model for a system with one stack could be polled using the following command:

### Example with BMS IP address of 192.168.1.21

```
modpoll.exe -m tcp -0 -r 40094 -c 22 192.168.1.21

modpoll 3.4 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2013 proconX Pty Ltd
Visit http://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration...: address = 1, start reference = 40094 (PDU), count = 22
```

```
Communication.....: 192.168.1.21, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table
```

```
-- Polling slave... (Ctrl-C to stop)
```

```
[40094]: 802
```

```
[40095]: 20
```

```
.
```

```
.
```

```
.
```

```
[40114]: -2
```

```
[40115]: -32768
```

## 8.2. CAN Bus Protocol Support



The details below do not apply to no-CAN variants (\*-NC) as they do not have a CAN bus interface.

Nuvation Energy BMS uses a flexible CAN reporting implementation which maps BMS software registers to CAN message identifiers. It is based on the CAN 2.0 standard, with the exception that Remote Transmission Requests (RTR) are not supported.

The parameters for the CAN interface are:

- Baud: 500 kbit/s
- CAN ID: 11-bit Identifier (Base frame format)
- CAN payload length: variable from 1 byte to 8 bytes based on register size

### 8.2.1. CAN Reporting and Commands

The CAN interface provides the following capabilities through configuration:

- Reporting of up to 64 individual registers within the Battery Management System.
- Bulk-reporting of registers that represent sets of measurements (e.g. cell voltages and temperatures)
- Creation of customizable commands to the Battery Management System to implement actions such as:
  - connecting/disconnecting the battery to the DC bus
  - clearing faults and/or warnings
  - updating the controller heartbeat (i.e. watchdog)

These capabilities are described in [Section 7.8.1, "CAN Bus"](#).

### 8.2.2. Data Format



The protocol is based on the CAN message frames. A reference for the CAN message frames can be found at [https://en.wikipedia.org/wiki/CAN\\_bus](https://en.wikipedia.org/wiki/CAN_bus). All register data is transmitted as little endian (least significant byte first).

### 8.2.3. Data Size

The size of the register data transmitted/received in a CAN frame will depend of the *variable type* of the register. The *type* and corresponding *variable type* for all registers used in Battery Management System can be found in [Section 7.1.2, "Units"](#). For example a variable type of int32 represents a signed 32 bit integer and is 4 bytes in length. All register types can be read from the Operator Interface *Registers* menu option as discussed in [Section 6.3.1, "Registers"](#)

## 9. External Interfaces

### 9.1. High-Voltage Stack Controller

#### 9.1.1. Ethernet

The Ethernet jack is a standard RJ45 Cat5e rated jack.

**Table 34. Ethernet Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	TD_P	Transmit differential pair positive	External Equipment
2	TD_N	Transmit differential pair negative	External Equipment
3	RD_P	Receive differential pair positive	External Equipment
4	NUL45	Unused; connected to Pin 5 and terminated	External Equipment
5	NUL45	Unused; connected to Pin 4 and terminated	External Equipment
6	RD_N	Receive differential pair negative	External Equipment
7	NUL78	Unused; connected to Pin 8 and terminated	External Equipment
8	NUL78	Unused; connected to Pin 7 and terminated	External Equipment

#### 9.1.2. CAN 485



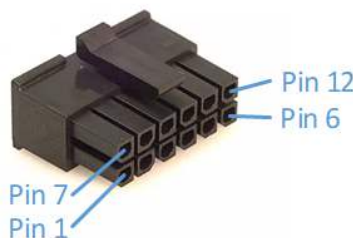
The details below do not apply to no-CAN variants (\*-NC) as they do not have a CAN bus interface. The RS-485 (Modbus) details continue to apply as it is available in the no-CAN variant.

This interface provides an isolated CAN 2.0 port and a non-isolated RS-485 (Modbus) port.

120  $\Omega$  termination for CAN is added by connecting pins 3 and 9 together. 150  $\Omega$  termination for Modbus is added by connecting pins 6 and 12 together.

Keep the termination wire length short for best results.

**Table 35. CAN 485: Molex Micro-Fit 3.0 Connector**



Molex 43025-1200	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	43025-1200

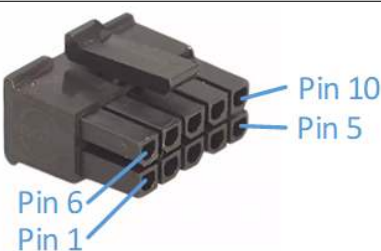
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	12
<b>Crimp terminal</b>	43030-0002
<b>Wire gauge range</b>	AWG20–24 stranded

**Table 36. CAN 485 Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	-V_isoCAN	Power return from Pin 7	External Equipment
2	CAN_N	CAN bus differential pair negative	External Equipment
3	EXTCAN_TERM1	Termination Resistor; short to Pin 9 to add 120Ω bus termination	CAN 485 Connector
4	COM	Power return from Stack Controller	External Equipment
5	MODBUS_N	MODBUS differential pair negative	External Equipment
6	EXTMOD_TERM1	Termination Resistor; short to Pin 12 to add 150Ω bus termination	CAN 485 Connector
7	+12V_isoCAN	+5.5-12 V DC isolated CAN bus power	External Equipment
8	CAN_P	CAN bus differential part positive	External Equipment
9	EXTCAN_TERM2	Termination Resistor; short to Pin 3 to add 120Ω bus termination	CAN 485 Connector
10	+VSYS	+24Vdc Power supply	External Equipment
11	MODBUS_P	MODBUS differential pair positive	External Equipment
12	EXTMOD_TERM2	Termination Resistor; short to Pin 6 to add 150Ω bus termination	CAN 485 Connector

## 9.1.3. GPIO-Out

**Table 37. GPO: Molex Micro-Fit 3.0 Connector**



Molex 43025-1000	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	43025-1000
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	10
<b>Crimp terminal</b>	43030-0002

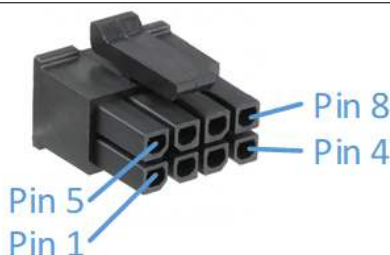
<b>Wire gauge range</b>	AWG20–24 stranded
-------------------------	-------------------

**Table 38. GPO Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	GPO_ISO0_A	Digital Output 0	External Equipment
2	GPO_ISO1_A	Digital Output 1	External Equipment
3	GPO_ISO2_A	Digital Output 2	External Equipment
4	GPO_ISO3_A	Digital Output 3	External Equipment
5	COM	Power return from Stack Controller	External Equipment
6	GPO_ISO0_B	Digital Output 0	External Equipment
7	GPO_ISO1_B	Digital Output 1	External Equipment
8	GPO_ISO2_B	Digital Output 2	External Equipment
9	GPO_ISO3_B	Digital Output 3	External Equipment
10	No Connect	Not Connected	No Connect

## 9.1.4. GPIO-In

**Table 39. GPI: Molex Micro-Fit 3.0 Connector**



Molex 43025-0800	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	43025-0800
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	8
<b>Crimp terminal</b>	43030-0002
<b>Wire gauge range</b>	AWG20–24 stranded

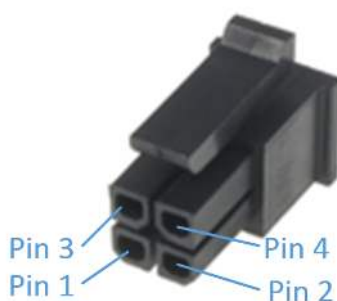
**Table 40. GPI Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	COM	Power return from Stack Controller for GPIO	External Equipment
2	COM	Power return from Stack Controller for GPI1	External Equipment

Pin	Connection	Description	Connected to Device
3	COM	Power return from Stack Controller for GPI2	External Equipment
4	COM	Power return from Stack Controller for GPI3	External Equipment
5	GPI_ISO0_K	Input detector 0	External Equipment
6	GPI_ISO1_K	Input detector 1	External Equipment
7	GPI_ISO2_K	Input detector 2	External Equipment
8	GPI_ISO3_K	Input detector 3	External Equipment

## 9.1.5. Link Out

**Table 41. Link Out: Molex Micro-Fit 3.0 Connector**



Molex 43025-0400	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	43025-0400
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	4
<b>Crimp terminal</b>	43030-0002
<b>Wire gauge range</b>	AWG20–24 stranded

**Table 42. Link Out Connector Pin Assignment**

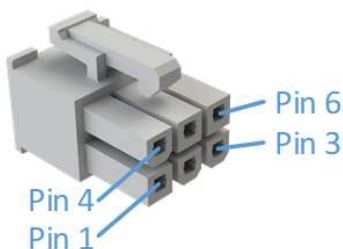
Pin	Connection	Description	Connected to Device
1	VBUS	DC power from Stack Controller, with Fault Pilot Signal	Cell Interface
2	COM	Power return from Stack Controller	Cell Interface
3	IPA	Link Bus differential pair plus	Cell Interface
4	IMA	Link Bus differential pair minus	Cell Interface

## 9.1.6. Stack Bus

The Stack Bus connector is a 6-pin Mini-Fit<sup>®</sup> Jr. Molex connector.

120  $\Omega$  termination must also be added by connecting pins 1 and 4 together with a short length of wire.

**Table 43. Stack Bus: Molex MiniFit Jr Connector**



Molex 39-01-2065	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	39-01-2065
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	6
<b>Crimp terminal</b>	39-00-0181
<b>Wire gauge range</b>	AWG20–24 stranded

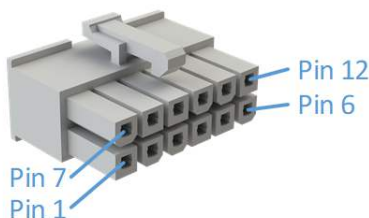
**Table 44. Stack Bus Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	TERM1	Termination Resistor; Short to Pin 4 to add 120 $\Omega$ bus termination	Stack Bus Connector
2	STACKBUS_N	Stack bus differential pair negative	Power Interface
3	+VSYS	+24V Power Supply	Power Interface
4	TERM2	Termination Resistor; Short to Pin 1 to add 120 $\Omega$ bus termination	Stack Bus Connector
5	STACKBUS_P	Stack bus differential pair positive	Power Interface
6	COM	Power return from Stack Controller	Power Interface

## 9.2. High-Voltage Power Interface

### 9.2.1. Contactors

**Table 45. Contactors: Molex MiniFit Jr Connector**



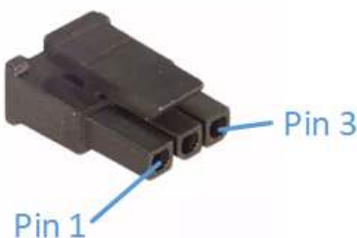
Molex 39-01-2125	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	39-01-2125
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	12
<b>Crimp terminal</b>	39-00-0181
<b>Wire gauge range</b>	AWG18–24 stranded

**Table 46. Contactor Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	COIL1_HI	Positive Coil 1	Contactor 1 positive coil connection
2	COIL2_HI	Positive Coil 2	Contactor 2 positive coil connection
3	COIL3_HI	Positive Coil 3	Contactor 3 positive coil connection
4	COIL4_HI	Positive Coil 4	Contactor 4 positive coil connection
5	No Connect	Not Connected	No Connect
6	COM	Negative reference for external supply	External Power Supply
7	COM	Negative Coil 1	Contactor 1 negative coil connection
8	COM	Negative Coil 2	Contactor 2 negative coil connection
9	COM	Negative Coil 3	Contactor 3 negative coil connection
10	COM	Negative Coil 4	Contactor 4 negative coil connection
11	+VINT	Power Interface Power Supply	Connect to Contactors connector Pin 12 if driving contactor coil from Power Interface power supply
12	+VCOIL	12-24V Contactor Coil Power Supply	Connect to external power supply or Contactors connector Pin 11 if driving contactor coil from Power Interface power supply

## 9.2.2. Interlock

**Table 47. Interlock: Molex Micro-Fit Connector**



Molex 43645-0300	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	43645-0300
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	3
<b>Crimp terminal</b>	43030-0002
<b>Wire gauge range</b>	AWG20–24 stranded

**Table 48. Interlock Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	R_OVERRIDE_ENA#	Active-low; allows Nuvation Energy BMS software to control contactors	Interlock Connector Pin 3
2	R_DRV_ENA#	Active-low; allows internal hardware fault detection to override Nuvation Energy BMS software control of contactors	Interlock Connector Pin 3
3	COM	Power return from Power Interface	Interlock Connector Pin 1 or Pin 2

## 9.2.3. External Power

The external supply can be either 16--24 V AC or 13–34 V DC and must be isolated from the battery stack, chassis, and COM grounds.

There is no polarity dependency between PWR\_A and PWR\_B pins.

**Table 49. External Power: Molex MiniFit Jr Connector**





Molex 39-01-2025	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	39-01-2025
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	2
<b>Crimp terminal</b>	39-00-0181
<b>Wire gauge range</b>	AWG18–24 stranded

**Table 50. External Power Connector Pin Assignment**

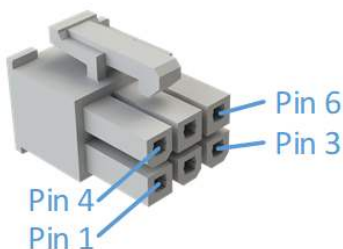
Pin	Connection	Description	Connected to Device
1	PWR_A	External Power Supply Input	External Power Supply
2	PWR_B	External Power Supply Input	External Power Supply

## 9.2.4. Stack Bus

The Stack Bus provides 42 mA to the Stack Controller plus the summation of current consumed by all Cell Interface modules in the system (up to 25 mA per CI-12 or 31 mA per CI-16/CI-4M12).

120  $\Omega$  termination must be added by connecting pins 1 and 3 together with a short length of wire.

**Table 51. Stack Bus: Molex MiniFit Jr Connector**



Molex 39-01-2065	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	39-01-2065
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	6
<b>Crimp terminal</b>	39-00-0181
<b>Wire gauge range</b>	AWG18–24 stranded

**Table 52. Stack Bus Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	TERM1	Termination Resistor; Short to Pin 4 to add 120 $\Omega$ bus termination	Stack Bus Connector
2	STACKBUS_N	Stack bus differential pair negative	Stack Controller
3	+VSYS	+24V Power Supply	Stack Controller
4	TERM2	Termination Resistor; Short to Pin 1 to add 120 $\Omega$ bus termination	Stack Bus Connector
5	STACKBUS_P	Stack bus differential pair positive	Stack Controller
6	COM	Power return from Power Interface	Stack Controller

## 9.2.5. Thermistor

The thermistor must be electrical isolated from the battery stack. Contact Nuvation Energy for support if temperature compensation of the high-current shunt is desired for your specific application.



Nuvation Energy can be contacted via [support@nuvationenergy.com](mailto:support@nuvationenergy.com).

## 9.2.6. Current Shunt

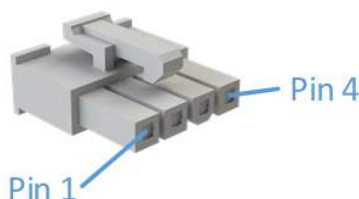
For best results, minimize the cable length used between the shunt and the connector. Use a twisted pair for the differential shunt voltage sense wires.

The differential voltage across the shunt must never exceed 1V under any circumstance. Choose the resistance according to the application.



Full stack potential exists between the 'Stack Power' and the 'Current Shunt' connectors. Potentially dangerous voltages may also be present between either or both of these connectors and earth or chassis ground, including the Power Interface and Stack Controller enclosures. Appropriate safety precautions must be observed.

**Table 53. Current Shunt: Molex MiniFit Jr Connector**



Molex 39-01-4041	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	39-01-4041
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	4
<b>Crimp terminal</b>	39-00-0181
<b>Wire gauge range</b>	AWG18–24 stranded

**Table 54. Current Shunt Connector Pin Assignment**

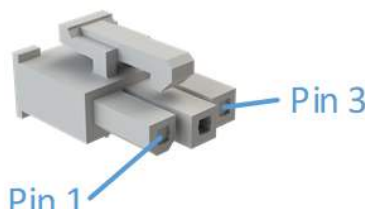
Pin	Connection	Description	Connected to Device
1	No Connect	Not Connected	No Connect
2	VSHUNT_REF	Voltage reference for voltage measurement	Load side of current shunt
3	VSHUNT_LOAD	Differential voltage input; Load side	Load side of current shunt
4	VSHUNT_BAT	Differential voltage input; Battery side	Battery side of current shunt

## 9.2.7. Stack Power



Full stack potential exists between the 'Stack Power' and the 'Current Shunt' connectors. Potentially dangerous voltages may also be present between either or both of these connectors and earth or chassis ground, including the Power Interface and Stack Controller enclosures. Appropriate safety precautions must be observed.

**Table 55. Stack Power: Molex MiniFit Jr Connector**



Molex 39-01-4031	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	39-01-4031
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	3

<b>Crimp terminal</b>	39-00-0181
<b>Wire gauge range</b>	AWG18–24 stranded

**Table 56. Stack Power Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	+VBAT_POS	Overall Stack Positive	Connect to most positive terminal of the battery stack
2	No Connect	Not Connected	No Connect
3	No Connect	Not Connected	No Connect

## 9.2.8. Reset Push-Button

The reset push-button is accessible through a small hole in the enclosure. It resets the Power Interface and cycles power on the Stack Bus, thereby resetting the Stack Controller and all Cell Interface modules powered by the Link Bus. The button needs to be held for 0.5 seconds to issue the reset.

## 9.3. Cell Interface

### 9.3.1. Battery Cells

The Battery Cells connector provides cell voltage input and a means for balancing the cells. The cable wire should be rated for at least 750 mA to survive worse-case currents.

Pins 8, 16, 17, and 18 are No Connect in the Cell Interface - 12 channel model. Pins 2, 4, 6, 8, 10, 11, 12, 13, 14, 15, 16, and 17 are No Connect in the Cell Interface - 12 V 4 channel model.

All unused voltage inputs should be tied to the next highest potential voltage sense input. In this way, all pins should be connected with the exception of pins 8, 16, 17 and 18 in the Cell Interface - 12 channel model and pins 2, 4, 6, 8, 10, 11, 12, 13, 14, 15, 16, and 17 in the Cell Interface - 12 V 4 channel model.



The wiring of the battery cell voltage and temperature sensing should be verified **before** connecting to the Cell Interface modules. The temperature sensing wires must be isolated from the cell voltage sensing leads. Although the Cell Interface includes protective circuitry to make it more resilient to brief wiring errors, the same circuitry can result in the battery cells being slowly discharged. Over time, these wiring errors can cause damage to the Cell Interface and/or the cells.

**Table 57. Battery Cells: Molex Micro-Fit 3.0 Connector**



Molex 43025-1800	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	43025-1800
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	18
<b>Crimp terminal</b>	43030-0002
<b>Wire gauge range</b>	AWG20–24 stranded

#### 9.3.1.1. Battery Cell Connector for Cell Interface - 12 channel

**Table 58. Cell Interface - 12 channel Battery Cell Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	CELL0	Bottom reference of Cell Interface	Connect to negative terminal of the lowest cell (Cell 1)
2	CELL2	Cell 2 voltage sense	Connect to positive terminal of Cell 2
3	CELL4	Cell 4 voltage sense	Connect to positive terminal of Cell 4

Pin	Connection	Description	Connected to Device
4	CELL6	Cell 6 voltage sense	Connect to positive terminal of Cell 6
5	CELL8	Cell 8 voltage sense	Connect to positive terminal of Cell 8
6	CELL10	Cell 10 voltage sense	Connect to positive terminal of Cell 10
7	CELL12	Cell 12 voltage sense	Connect to positive terminal of Cell 12
8	No Connect	Not Connected	No Connect
9	No Connect	Not Connected	No Connect
10	CELL1	Cell 1 voltage sense	Connect to positive terminal of the lowest cell (Cell 1)
11	CELL3	Cell 3 voltage sense	Connect to positive terminal of Cell 3
12	CELL5	Cell 5 voltage sense	Connect to positive terminal of Cell 5
13	CELL7	Cell 7 voltage sense	Connect to positive terminal of Cell 7
14	CELL9	Cell 9 voltage sense	Connect to positive terminal of Cell 9
15	CELL11	Cell 11 voltage sense	Connect to positive terminal of Cell 11
16	No Connect	Not Connected	No Connect
17	No Connect	Not Connected	No Connect
18	No Connect	Not Connected	No Connect

## 9.3.1.2. Battery Cell Connector for Cell Interface - 16 channel

**Table 59. Cell Interface - 16 channel Battery Cell Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	CELL0	Bottom reference of Cell Interface	Connect to negative terminal of the lowest cell (Cell 1)
2	CELL2	Cell 2 voltage sense	Connect to positive terminal of Cell 2
3	CELL4	Cell 4 voltage sense	Connect to positive terminal of Cell 4
4	CELL6	Cell 6 voltage sense	Connect to positive terminal of Cell 6
5	CELL8	Cell 8 voltage sense	Connect to positive terminal of Cell 8
6	CELL10	Cell 10 voltage sense	Connect to positive terminal of Cell 10
7	CELL12	Cell 12 voltage sense	Connect to positive terminal of Cell 12
8	CELL14	Cell 14 voltage sense	Connect to positive terminal of Cell 14
9	No Connect	Not Connected	No Connect
10	CELL1	Cell 1 voltage sense	Connect to positive terminal of the lowest cell (Cell 1)
11	CELL3	Cell 3 voltage sense	Connect to positive terminal of Cell 3
12	CELL5	Cell 5 voltage sense	Connect to positive terminal of Cell 5
13	CELL7	Cell 7 voltage sense	Connect to positive terminal of Cell 7
14	CELL9	Cell 9 voltage sense	Connect to positive terminal of Cell 9
15	CELL11	Cell 11 voltage sense	Connect to positive terminal of Cell 11
16	CELL13	Cell 13 voltage sense	Connect to positive terminal of Cell 13

Pin	Connection	Description	Connected to Device
17	CELL15	Cell 15 voltage sense	Connect to positive terminal of Cell 15
18	CELL16	Cell 16 voltage sense	Connect to positive terminal of Cell 16

### 9.3.1.3. Battery Cell Connector for Cell Interface - 12 V 4 channel

**Table 60. Cell Interface - 12 V 4 channel Battery Cell Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	BLOCK0	Bottom reference of Cell Interface	Connect to negative terminal of the lowest cell (Cell 1)
2	No Connect	Not Connected	No Connect
3	BLOCK1	Cell 1 voltage sense	Connect to positive terminal of the lowest cell (Cell 1)
4	No Connect	Not Connected	No Connect
5	BLOCK2	Cell 2 voltage sense	Connect to positive terminal of Cell 2
6	No Connect	Not Connected	No Connect
7	BLOCK3	Cell 3 voltage sense	Connect to positive terminal of Cell 3
8	No Connect	Not Connected	No Connect
9	No Connect	Not Connected	No Connect
10	No Connect	Not Connected	No Connect
11	No Connect	Not Connected	No Connect
12	No Connect	Not Connected	No Connect
13	No Connect	Not Connected	No Connect
14	No Connect	Not Connected	No Connect
15	No Connect	Not Connected	No Connect
16	No Connect	Not Connected	No Connect
17	No Connect	Not Connected	No Connect
18	BLOCK4	Cell 4 voltage sense	Connect to positive terminal of Cell 4

## 9.3.2. Temperature Sensors

The Temperature Sensors connector provides NTC thermistor inputs for temperature measurement of the cells and/or surrounding area. Because the sensors are referenced to the CELL0 (or BLOCK0 on CI-4M12) input on the Cell Interface, care must be taken to ensure that they are electrically isolated from any common or ground potential, and from all other cell voltage terminals of all Cell Interface modules in the system. The thermistors must be isolated from the cell voltage terminals in such a way that they will not make an electrical connection to a cell terminal in the event of vibration/failures.

Connect up to eight 10 kΩ NTC thermistors to this connector.



2 to 7 thermistors are supported to accommodate systems requiring Nuvation Energy BMS to meet functional safety requirements.

**Table 61. Temperature Sensors: Molex Micro-Fit 3.0 Connector**



Molex 43025-1600	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	43025-1600
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	16
<b>Crimp terminal</b>	43030-0002
<b>Wire gauge range</b>	AWG20–24 stranded

**Table 62. Temperature Sensors Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	VBOT	External Temperature Probe Reference 1	10 kΩ NTC Thermistor
2	VBOT	External Temperature Probe Reference 2	10 kΩ NTC Thermistor
3	VBOT	External Temperature Probe Reference 3	10 kΩ NTC Thermistor
4	VBOT	External Temperature Probe Reference 4	10 kΩ NTC Thermistor
5	VBOT	External Temperature Probe Reference 5	10 kΩ NTC Thermistor
6	VBOT	External Temperature Probe Reference 6	10 kΩ NTC Thermistor
7	VBOT	External Temperature Probe Reference 7	10 kΩ NTC Thermistor
8	VBOT	External Temperature Probe Reference 8	10 kΩ NTC Thermistor
9	TEMP1_R	External Temperature Probe Input 1	10 kΩ NTC Thermistor



Pin	Connection	Description	Connected to Device
10	TEMP2_R	External Temperature Probe Input 2	10 kΩ NTC Thermistor
11	TEMP3_R	External Temperature Probe Input 3	10 kΩ NTC Thermistor
12	TEMP4_R	External Temperature Probe Input 4	10 kΩ NTC Thermistor
13	TEMP5_R	External Temperature Probe Input 5	10 kΩ NTC Thermistor
14	TEMP6_R	External Temperature Probe Input 6	10 kΩ NTC Thermistor
15	TEMP7_R	External Temperature Probe Input 7	10 kΩ NTC Thermistor
16	TEMP8_R	External Temperature Probe Input 8	10 kΩ NTC Thermistor



For safety certified applications there must be at least 2—but no more than 7—thermistors installed per Cell Interface module in a specific pattern. The following table lists the expected pattern of thermistor installation approved for safety certification.

**Table 63. Safety Certified Thermistor Installation**

Total Thermistors per Cell Interface	TEMP1_R	TEMP2_R	TEMP3_R	TEMP4_R	TEMP5_R	TEMP6_R	TEMP7_R	TEMP8_R
7	Installed	Installed	Installed	Installed	Installed	Installed	Installed	Not Installed
6	Installed	Installed	Installed	Installed	Not Installed	Installed	Installed	Not Installed
5	Installed	Installed	Not Installed	Installed	Not Installed	Installed	Installed	Not Installed
4	Not Installed	Installed	Installed	Installed	Installed	Not Installed	Not Installed	Not Installed
3	Not Installed	Installed	Installed	Installed	Not Installed	Not Installed	Not Installed	Not Installed
2	Not Installed	Installed	Installed	Not Installed	Not Installed	Not Installed	Not Installed	Not Installed

### 9.3.2.1. Thermal Consistency

For safety certified systems, there is an additional constraint on thermal consistency for all temperature measurements for each Cell Interface. The constraint is described in detail in [Section 7.10.9, “Temperature Consistency Check”](#).

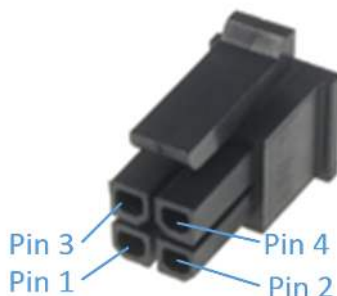
### 9.3.3. Link In

This interface is used, with Link Bus cables, to connect each Cell Interface in series to the Cell Interface of the next lowest potential, and the Cell Interface of the lowest potential to the Stack Controller. In this series-connected chain of modules, each connection is made from the Link In connector of the module with higher potential to the Link Out connector of the module with lower potential.

The Cell Interface also accepts power from its Link In connector to power itself, as well as subsequent Cell Interface modules in the Link Bus chain (i.e. connected to its Link Out connector). The amount of current sourced into this connector is the sum of current consumed by this Cell Interface and the subsequent Cell Interface modules in the Link Bus chain (up to 25 mA per CI-12 or 31 mA per CI-16 /CI-4M12 ).

Connect this interface to the [Link Out connector](#) on the previous Cell Interface module in the Link Bus chain or to the [Link Out connector](#) on the Stack Controller.

**Table 64. Link In: Molex Micro-Fit 3.0 Connector**



Molex 43025-0400	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	43025-0400
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	4
<b>Crimp terminal</b>	43030-0002
<b>Wire gauge range</b>	AWG20–24 stranded

**Table 65. Link In Connector Pin Assignment**

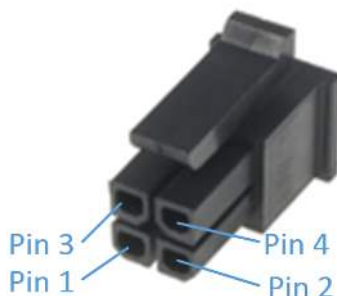
Pin	Connection	Description	Connected to Device
1	VBUS	DC power from the Stack Controller, with Fault Pilot Signal	Previous Cell Interface in the Link Bus chain or Stack Controller
2	COM	Power return from the Stack Controller	Previous Cell Interface in the Link Bus chain or Stack Controller
3	IPA	Link Bus differential pair plus	Previous Cell Interface in the Link Bus chain or Stack Controller
4	IMA	Link Bus differential pair minus	Previous Cell Interface in the Link Bus chain or Stack Controller

## 9.3.4. Link Out

This interface is used to provide a data channel and power source to subsequent Cell Interface modules in the Link Bus chain. The amount of current supplied by this connector is the sum of current consumed by all subsequent Cell Interface modules in the Link Bus chain (up to 25 mA per CI-12 or 31 mA per CI-16/CI-4M12).

Connect this interface to the [Link In connector](#) on the subsequent Cell Interface module in the Link Bus chain. The Link Out interface of the last Cell Interface module in the Link Bus chain is to be left unconnected.

**Table 66. Link Out: Molex Micro-Fit 3.0 Connector**



Molex 43025-0400	
<b>Manufacturer</b>	Molex Incorporated
<b>Housing</b>	43025-0400
<b>Housing material</b>	Nylon UL94V-0
<b>Circuits</b>	4
<b>Crimp terminal</b>	43030-0002
<b>Wire gauge range</b>	AWG20-}24 stranded

**Table 67. Link Out Connector Pin Assignment**

Pin	Connection	Description	Connected to Device
1	VBUS	DC power from Stack Controller , with Fault Pilot Signal	Subsequent Cell Interface module in the Link Bus chain
2	COM	Power return from Stack Controller	Subsequent Cell Interface module in the Link Bus chain
3	IPA	Link Bus differential pair plus	Subsequent Cell Interface module in the Link Bus chain
4	IMA	Link Bus differential pair minus	Subsequent Cell Interface module in the Link Bus chain

## 10. Troubleshooting

### 10.1. Faults and Self Check Failures

During initial setup and configuration of a Nuvation Energy BMS, there are two main classes of issues that can be encountered:

1. Self check issues
2. Triggering of faults

The first class of issues are encountered during initial deployment of a Nuvation Energy BMS, particularly while attempting to power on the Battery Management System or exit Service Lockout. When this event occurs, an operator of the Operator Interface can open the Details > Safety accordion ([Section 6.2.3, "Safety"](#)). That screen presents a list of faults and warnings that may be either *clear* or *checking*. When all faults/warnings are clear, the Battery Management System become operational. All faults/warnings that are in a *checking* state indicate that the Self Check operation is still executing. The data that is associated with that fault/warning is not being received.

This section will provide details on why the faults/warnings are failing their self check. Once all faults/warnings are clear, Nuvation Energy BMS has been configured and installed as expected (i.e. all inputs are receiving data). When this occurs the software enters its operational mode.

The second class of issues occurs while Nuvation Energy BMS is operational. One or more faults can be triggered causing Nuvation Energy BMS to exit its operating condition and open all contactors. There are many faults that can be triggered. This section will describe the condition the fault monitors and the meaning when that fault is triggered.

The following sections describe different classes of faults/warnings and issues surrounding self check failures. In general, all warnings have a similar trigger condition as their corresponding fault. The following discussion will focus on the term fault and all descriptions can be applied to the compatible warning.

Each fault has a descriptive name (e.g. Cell Over Voltage) and a name for the corresponding trigger component (e.g. `stack_fault_cell_over`). Both names are shown on the details tab of the OI, refer to [Section 6.2.3, "Safety"](#) for details.

#### 10.1.1. Cell Voltage Faults

##### **Cell Over Voltage** | `stack_fault_cell_over`

- Fault that is triggered when an installed cell voltage is above the *over* fault threshold

##### **Cell High Voltage** | `stack_fault_cell_hi`

- Fault that is triggered when an installed cell voltage is above the *high* fault threshold

##### **Cell Low Voltage** | `stack_fault_cell_lo`

- Fault that is triggered when an installed cell voltage is below the *low* fault threshold

##### **Cell Under Voltage** | `stack_fault_cell_under`

- Fault that is triggered when an installed cell voltage is below the *under* fault threshold

## 10.1.1.1. Self Check Issues

These faults could fail self check through a number of possible conditions such as:

1. Missing or misconfigured Cell Interface modules, preventing the software from scanning for the measurements.
2. Misconfiguration on the number of Cell Interface modules. Refer to the register `sc_linkbus.cicount`.
3. Incorrect Cell Interface configuration. For example if the type of Cell Interface defined in `sc_linkbus.softwareid` is incorrect, it may prevent the cell voltage measurement.
4. Failure in the Link Bus communication. Any interruptions of the bus can prevent the cell voltages from completing a self check. Such a failure could happen if:
  - Not all Cell Interface modules are connected
  - Link Bus cables are connected to the wrong port (i.e. Link Out instead of Link In).
  - Link Bus cables are damaged

## 10.1.2. Stack Voltage Faults

### Stack Over Voltage | `stack_fault_voltage_over`

- Fault that is triggered when the stack voltage is above the *over* fault threshold.

### Stack High Voltage | `stack_fault_voltage_hi`

- Fault that is triggered when the stack voltage is above the *high* fault threshold.

### Stack Low Voltage | `stack_fault_voltage_lo`

- Fault that is triggered when the stack voltage is below the *low* fault threshold.

### Stack Under Voltage | `stack_fault_voltage_under`

- Fault that is triggered when the stack voltage is below the *under* fault threshold.

### Voltage Sum Check | `stack_fault_voltage_sum`

- Fault that is triggered when the absolute difference between the measured stack voltage and sum of all cell voltages in the stack exceeds the fault threshold.

## 10.1.2.1. Self Check Issues

These faults have the following self check issues:

Fault	Self Check Issues
<code>stack_fault_voltage_hi</code>	Power Interface AFE is configured to be disabled
<code>stack_fault_voltage_lo</code>	
<code>stack_fault_voltage_sum</code>	Same issues regarding cell voltage self check issues ( <a href="#">Section 10.1.1.1, "Self Check Issues"</a> )

## 10.1.3. Thermal Faults

**Discharge Under Temperature | `stack_fault_discharge_therm_over`**

- Fault that is triggered when any thermistor measurement is above the *over* fault threshold during discharge

**Discharge High Temperature | `stack_fault_discharge_therm_hi`**

- Fault that is triggered when any thermistor measurement is above the *high* fault threshold during discharge

**Discharge Low Temperature | `stack_fault_discharge_therm_lo`**

- Fault that is triggered when any thermistor measurement is below the *low* fault threshold during discharge

**Discharge Under Temperature | `stack_fault_discharge_therm_under`**

- Fault that is triggered when any thermistor measurement is below the *under* fault threshold during discharge

**Charge Over Temperature | `stack_fault_charge_therm_over`**

- Fault that is triggered when any thermistor measurement is above the *over* fault threshold during charge

**Charge High Temperature | `stack_fault_charge_therm_hi`**

- Fault that is triggered when any thermistor measurement is above the *high* fault threshold during charge

**Charge Low Temperature | `stack_fault_charge_therm_lo`**

- Fault that is triggered when any thermistor measurement is below the *low* fault threshold during charge

**Charge Under Temperature | `stack_fault_charge_therm_under`**

- Fault that is triggered when any thermistor measurement is below the *under* fault threshold during charge

## 10.1.3.1. Self Check Issues

All of the temperature measurements are communicated through over the Link Bus using the same mechanism as the cell voltages. Thus, the self check failures for the temperature measurements are exactly the same. Refer to [Section 10.1.1.1, "Self Check Issues"](#) for further details.

## 10.1.4. Stack Current Faults

**Discharge Over Current | `stack_fault_discharge_current_over`**

- Fault that is triggered when the stack current is above the *over* discharge fault threshold.

**Discharge High Current | `stack_fault_discharge_current_hi`**

- Fault that is triggered when the stack current is above the *high* discharge fault threshold.

**Charge High Current | `stack_fault_charge_current_hi`**

- Fault that is triggered when any stack current is above the *high* charge fault threshold.

**Charge Over Current | `stack_fault_charge_current_over`**

- Fault that is triggered when any stack current is above the *over* charge fault threshold.



Both of these prior faults could trigger if the current shunt is not properly connected or has not been configured for its proper measurement range.

## 10.1.4.1. Self Check Issues

The stack current is measured from the Power Interface Analog Front End (AFE). The following issues can contribute to these faults failing their self check:

1. Power Interface AFE is not enabled
2. Stack bus failure between the Stack Controller and Power Interface

## 10.1.5. Pre-charge Faults

### **Pre-charge Timeout** | `stack_fault_precharge_timeout`

- Fault that is triggered at the end of the pre-charge period if either:
  1. the delay set by `stack_control.precharge_delay` expired before a current is measured, or
  2. the measured current exceeds the maximum pre-charge current
- The value of `stack_control.precharge_error` can be used to determine the reason for triggering the fault

### **Pre-charge Over Current** | `stack_fault_precharge_over_current`

- Fault that is triggered at any time during the pre-charge connection period if the stack current exceeds the fault threshold

## 10.1.5.1. Self Check Issues

These faults are directly related to the current faults detailed previously. Refer to [Section 10.1.4.1, "Self Check Issues"](#) for details.

## 10.1.6. Contactor Faults

### **Contactor Coil Drive Check** | `stack_fault_coil_fail`

- Fault that is generated when there is a difference between the commanded state of the contactor coil and the observed state read from the hardware. This fault is a consistency check on the drive state of the coil. This fault will trigger when contactors are unconnected or have a short in their circuit. If the fault pilot signal is asserted, this fault will trigger and can not be used to determine if there is a inconsistent contactor drive state.

### **Contactor Feedback Check** | `stack_fault_contactor_feedback_fail`

- Fault that is generated when there is a mismatch between the contactor state and the contactor feedback signal provided. Note that the feedback is provided through a GPI.

### **Interlock Check** | `sc_fault_pi_interlock`

- Fault that is generated when the interlock function of the High-Voltage BMS has been activated. The interlock when enabled is activated through the interlock circuit having an open condition (i.e. via an e-stop or door switch).

## Fault Pilot Check | `sc_fault_fault_pilot_state_mismatch`

- Fault that is generated when the fault pilot state set by the Stack Controller does not match the detected state on the Power Interface



`sc_fault_fault_pilot_state_mismatch.time_hyst` must be set to at least 500ms.

### 10.1.6.1. Self Check Issues

All of these faults are dependent on the stack bus communication between the Stack Controller and Power Interface modules. The following table summarizes these self check issues:

Fault	Self Check Issues
<code>sc_fault_pi_interlock</code>	Stack Bus failure
<code>sc_fault_fault_pilot_state_mismatch</code>	Stack Bus failure
<code>stack_fault_coil_fail</code>	Power Interface AFE disabled
<code>stack_fault_contactor_feedback_fail</code>	GPI not configured Power Interface AFE disabled Stack Bus failure

### 10.1.7. Diagnostics Faults

#### Cell Open Wire Test | `stack_fault_cell_open_wire`

- Fault indicating there is an open wire on an installed cell channel
- This can also trip due to misconfigured or incorrectly installed cells

#### Cell Open Wire Test | `stack_warn_cell_open_wire`

- Warning triggering on the same data as `stack_fault_cell_open_wire`

#### Thermistor Circuit Test | `stack_fault_therm_circuit_fail`

- Fault indicating there is a circuit failure on a thermistor channel on a Cell Interface
- This can also trip due to misconfigured or incorrectly installed thermistors

#### Thermistor Consistency Check | `stack_fault_ci_therm_consistency`

- Fault indicating the difference between thermal readings on a Cell Interface is too large

#### Cell Interface Measurement Check | `sc_fault_ci`

- Fault indicating there was an internal hardware failure on a Cell Interface

#### Current Shunt Open Wire Check | `stack_fault_open_shunt`

- Fault indicating the current shunt is open circuit

#### Current Shunt Short Circuit Check | `stack_fault_short_shunt`

- Fault indicating the current shunt is short circuit
- When experiencing difficulty clearing this fault, please refer to [Section 10.1.7.1, "Known Issues"](#).



## **Contactor End-Of-Life | ssg\_fault\_contactor\_end\_of\_life**

- Fault indicating a contactor is end-of-life
- Typically disabled

## **Stack Switchgear Contactor Life Sanity Check | ssg\_fault\_contactor\_life\_sanity\_check**

- Fault indicating a sanity check error on the contactor life data
- Typically disabled

## **Stack Switchgear Contactor Life Memory Error | ssg\_fault\_config\_contactor\_life**

- Fault indicating an error saving or loading the contactor life data
- Typically disabled

## **Stack Switchgear Contactor Life Memory Check | ssg\_fault\_config\_contactor\_life\_verify**

- Fault indicating a validation error on the contactor life data
- Typically disabled

## 10.1.7.1. Known Issues

### 10.1.7.1.1. Short Shunt Fault on Self Check

#### **Issue**

In rare occasions, the `stack_fault_short_shunt` can accidentally trip when performing a self check. If the contactors remain open and no current has flowed through the batteries, the fault does not indicate a short on the current shunt. Note that this issue only occurs within a minute of performing a self check and faults that happen outside this window should be investigated further.

#### **Workaround**

The operator of the Operator Interface can attempt to clear the fault as described in [Section 6.2.3, "Safety"](#).

### 10.1.7.1.2. Clearing the Short Shunt Fault Is Not Immediate

#### **Issue**

Clearing the `stack_fault_short_shunt` fault might not occur immediately.

#### **Workaround**

To determine when the fault can be cleared, read the values of the `stack_short_shunt_detector.voltage_activity_threshold` and `stack_short_shunt_detector.inactivity_envelope` registers. Wait until the value in the inactivity envelope is smaller than the voltage activity threshold. At this point the `stack_fault_short_shunt` fault can be cleared. The maximum time required for the inactivity envelope to fall below the threshold can be determined by reading the `stack_short_shunt_detector.activity_detection_time` register, with the value given in  $\mu\text{s}$ .

## 10.1.7.2. Self Check Issues

The `stack_fault_cell_open_wire`, `stack_fault_therm_circuit_fail`, `stack_fault_ci_therm_consistency`, and

sc\_fault\_ci can fail a self check due to problems with the Link Bus. Refer to [Section 10.1.1.1, "Self Check Issues"](#) for further details.

The stack\_fault\_open\_shunt and stack\_fault\_short\_shunt can fail a self check if the associated algorithms are not enabled, the Power Interface AFE is not enabled, or there is Stack Bus communication errors.

## 10.1.8. Breaker Faults

### **Breaker Tripped Check | stack\_fault\_breaker\_tripped**

- Fault that is triggered to indicate that the breaker has tripped (i.e. opened)

### **Breaker Conflict Check | stack\_fault\_breaker\_conflict**

- Fault that is triggered to indicate that the breaker state differs from the expected state of the breaker

### 10.1.8.1. Self Check Issues

Most self check failures for these faults are caused by an invalid configuration of the GPI(s) used to read the breaker state.

## 10.1.9. Watchdog Faults

### **Power Interface Measurement Watchdog | stack\_fault\_power\_wdt**

- Fault indicating stack voltage and current data has not been updated periodically

### **Cell Voltage Measurement Watchdog | stack\_fault\_cell\_wdt**

- Fault indicating all cell data has not been updated periodically for every installed cell

### **Temperature Measurement Watchdog | stack\_fault\_therm\_wdt**

- Fault indicating all thermistor data has not been updated periodically for every installed thermistor

### **Cell Interface Measurement Watchdog | stack\_fault\_ci\_wdt**

- Fault indicating Cell Interface data has not been updated periodically for every installed Cell Interface

### **Cell Open Wire Measurement Watchdog | stack\_fault\_open\_wire\_wdt**

- Fault indicating open wire scanning has not been occurring periodically when connected to the DC bus

### **Controller Heartbeat Watchdog | sc\_fault\_controller\_wdt**

- Fault indicating that an external controller to the Battery Management System was not updating its watchdog timer (via the MESA heartbeat)

### **Controller Heartbeat Watchdog | sc\_warn\_controller\_wdt**

- Warning triggered from the same data as sc\_fault\_controller\_wdt

### **StackBus Receive Watchdog | sc\_fault\_stackbus\_rxwdt**

- Fault indicating that there was a receive communication failure over the Stack Bus

**StackBus Transmit Watchdog | `sc_fault_stackbus_txwdt`**

- Fault indicating that there was a transmit communication failure over the Stack Bus

**Power Interface AFE Watchdog | `sc_fault_pi_afe_wdt`**

- Fault indicating that there was a communication failure between the AFE and the Power Interface

**LinkBus Communication Watchdog | `sc_fault_linkbus_wdt`**

- Fault indicating that there was an internal communication failure relating to Link Bus data. Please contact Nuvation Energy at [support@nuvationenergy.com](mailto:support@nuvationenergy.com) if you are experiencing this fault.

**Stack Controller RAM Test Watchdog | `sc_fault_ram_test_wdt`**

- Fault indicating the internal RAM testing has not been executing periodically

**Stack Controller ROM Test Watchdog | `sc_fault_rom_test_wdt`**

- Fault indicating the internal ROM testing has not been executing periodically

**Stack Controller Factory Memory Watchdog | `sc_fault_factory_verify_wdt`**

- Fault indicating the factory registers on the Stack Controller have not been periodically validated against non-volatile memory
- Typically disabled

**Power Interface Factory Memory Watchdog | `pi_fault_factory_verify_wdt`**

- Fault indicating the factory registers on the Power Interface have not been periodically being validated against non-volatile memory.
- Typically disabled

**Contactor Monitor Watchdog | `ssg_fault_contactor_monitor_wdt`**

- Fault indicating contactor monitoring has not been executing periodically
- Typically disabled

**Contactor End-Of-Life Watchdog | `ssg_fault_contactor_end_of_life_wdt`**

- Fault indicating contactor life has not been evaluated periodically
- Typically disabled

**Stack Switchgear Contactor Life Memory Watchdog | `ssg_fault_contactor_life_verify_wdt`**

- Fault indicating contactor life data has not been validated periodically
- Typically disabled

## 10.1.9.1. Self Check Issues

The `stack_fault_cell_wdt`, `stack_fault_therm_wdt`, `stack_fault_ci_wdt`, and `stack_fault_open_wire_wdt` can fail a self check due to problems with the Link Bus. Refer to [Section 10.1.1.1, "Self Check Issues"](#) for further details.

The `sc_fault_ram_test_wdt`, `sc_fault_rom_test_wdt`, `sc_fault_factory_verify_wdt`, `pi_fault_factory_verify_wdt`, `sc_fault_stackbus_rxwdt` and `sc_fault_stackbus_txwdt` should always pass their self check as they are core to the Battery Management System and does not have associated

configurability.

Please contact Nuvation Energy at [support@nuvationenergy.com](mailto:support@nuvationenergy.com) if you are experiencing these faults.

The `sc_fault_pi_afe_wdt` and `stack_fault_power_wdt` can fail a self check due to disabling the Power Interface AFE or a high amount of noise on the DC bus.

The `sc_fault_controller_wdt` can fail a self check if there is no external controller updating the heartbeat. In this case this fault should be disabled.

## 10.1.10. Miscellaneous Faults

### Stack Controller Configuration Consistency Check | `sc_fault_config`

- Fault indicating that there was an error reading the non-volatile storage of the Battery Management System configuration. A default configuration is used when this fault occurs and the Battery Management System will fail to exit Service Lockout. Once a configuration has been successfully imported into the Battery Management System and saved, this fault should not occur. Note that during an upgrade of the Battery Management System, the configuration is deleted to avoid incompatibility with the upgraded version.

### Stack Controller Factory Consistency Check | `sc_fault_config_factory`

- Fault indicating that there was an error reading the non-volatile storage of the Battery Management System factory configuration. This is similar to `sc_fault_config`; refer to that fault for more details.

### Stack Controller Factory Memory Check | `sc_fault_config_factory_verify`

- Fault indicating that verification of factory configuration settings against non-volatile storage has failed on the Stack Controller. This typically means either a setting was changed after saving or non-volatile memory has been corrupted.

### Power Interface Factory Memory Check | `pi_fault_config_factory_verify`

- Fault indicating that verification of factory configuration settings against non-volatile storage has failed on the Power Interface. This typically means either a factory setting was changed after saving or non-volatile memory has been corrupted.

### Firmware Version Mismatch Check | `sc_fault_fw_mismatch`

- Fault indicating that the Stack Controller and Power Interface firmware versions are different. Perform an upgrade to the desired Battery Management System firmware version.

### Clock Check | `sc_fault_clocks`

- Fault indicating that the Stack Controller and Power Interface clocks are ticking at different rates. This may indicate an overloaded system or malfunction in the hardware or software. Also if one of the processors resets, this fault can trip since its clock also resets.

### LinkBus Power Mode Check | `sc_fault_linkbus_power`

- Fault indicating that there was a hardware problem with the Link Bus power or an incorrect power mode in `sc_linkbus.power_mode`

### Internal Software Error | `sc_fault_register_link`

- Fault indicating an internal communication failure in the BMS firmware. Contact [support@nuvationenergy.com](mailto:support@nuvationenergy.com) if this failure occurs and record the values in the following

registers    `sc_gpo.link_error,`    `sc_gpo_shutdown.link_error,`    `stack_contactor.link_error`    and  
`sc_gpi.link_error`

## 10.2. Lost/Forgotten IP Address

If a Nuvation Energy BMS has been configured with a static IP address and it has been forgotten, follow the steps below to recover it.



Depending on the network interface used on the PC, this process may not work due to differing security and IP configurations. If the only IP discovered is the IP of the PC, the network interface is not suitable and another one will need to be used. This issue is most common with USB to Ethernet dongles.

### 10.2.1. Wireshark (Windows/Linux)

1. Download/install Wireshark on a PC (<https://www.wireshark.org/>)
2. Connect the PC directly to the Ethernet port on the High-Voltage BMS
3. Start a Wireshark capture on the network interface connected to the High-Voltage BMS
4. In the 'filter' field, enter in `arp.isgratuitous` and press enter
5. Either reboot the High-Voltage BMS, or unplug/plug the Ethernet cable
6. The device should send a 'Gratuitous ARP' on the Ethernet network. In Wireshark the 'Info' field looks like: Gratuitous ARP for <IP> (Request) where the <IP> is the address for the High-Voltage BMS
7. Once that is complete, update the PC network settings to match the High-Voltage BMS and connect the Operator Interface.
  - Refer to [Section 5.2, "Connect to the Operator Interface"](#) for instructions.

### 10.2.2. Netdiscover (Linux only)

1. Install netdiscover on a PC (on Debian based systems use: `sudo apt install netdiscover`)
2. Connect the PC directly to the Ethernet port on the High-Voltage BMS
3. Run `sudo netdiscover -i <interface> -p` where <interface> is the network interface connected to the High-Voltage BMS
4. Either reboot the High-Voltage BMS, or unplug/plug the Ethernet cable
5. The device address and MAC will show up in netdiscover once an ARP packet is sent
6. Once that is complete, update the PC network settings to match the High-Voltage BMS and connect the Operator Interface.
  - Refer to [Section 5.2, "Connect to the Operator Interface"](#) for instructions.



In the event the IP address cannot be discovered, a factory reset operation must be carried out to restore the BMS to its default IP configuration.

## Appendix A: Operating Limits

### High-Voltage Stack Controller



Exceeding the maximum ratings will damage the Stack Controller module.

### Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
<b>Stack Bus Specifications</b>						
+VSYS	Input Voltage	Supplied by Power Interface	5.6	24	34	V DC
	Input Current	+VSYS = 24 V DC	0.042	-	1.3	A DC
Rterm	Termination resistance tolerance	-	118.8	120	121.2	Ω
	Power rating	-	-	-	0.125	W
StackbusP	Dominant Output	-	2.45	-	3.3	V DC
	Recessive Output	-	-	2.3	-	V DC
	Output Current	-	10	-	50	mA DC
	Output Signal Rise Time	-	35	-	135	ns
	Output Signal Fall Time	-	35	-	135	ns
StackbusN	Dominant Output	-	0.5	-	1.25	V DC
	Recessive Output	-	-	2.3	-	V DC
	Output Current	-	10	-	50	mA DC
	Output Signal Rise Time	-	35	-	135	ns
	Output Signal Fall Time	-	35	-	135	ns
<b>Link Bus Specifications</b>						
+VBUS	Output Voltage	-	-	+VSYS	-	V DC
	Output Current	+VBUS = 24 V DC	-	-	1.26	A DC
IP_LINK	Output Current	-	-	-	20	mA DC
IN_LINK	Output Current	-	-	-	20	mA DC
<b>USB Specifications</b>						
+5V_USB	USB Current	-	-	-	500	mA DC
+5V_USB	USB Voltage	-	-	5	-	V DC
<b>Ethernet Specifications</b>						
ETH_Protocol	Ethernet data speeds	-	10	-	100	Base-T
ETH_Connector	Ethernet jack rating	-	-	Cat5e	-	
<b>GPIO-Out Specifications</b>						

Symbol	Parameter	Conditions	Min	Typ	Max	Units
V <sub>max</sub>	Open Blocking Voltage	Between *_A and *_B, or between *_B and *_A	-	-	60	V DC
I <sub>max</sub>	Closed Maximum Current	Between *_A and *_B, or between *_B and *_A	-	-	400	mA DC
R <sub>on</sub>	Closed-State Resistance	Between *_A and *_B, or between *_B and *_A	-	-	2	Ω
<b>GPIO-In Specifications</b>						
Turn-On	Turn On Threshold Voltage	-	0	-	1.4	V DC
	Turn-On Threshold Current	-	1.6	-	-	mA DC
Turn-Off	Turn-Off Threshold Voltage	-	3	-	5	V DC
	Turn-Off Threshold Current	-	-	1	-	mA DC
V <sub>max</sub>	Off Voltage	I <sub>in</sub> = 0 mA	-	-	5	V DC
I <sub>max</sub>	On Current	V <sub>in</sub> = 0 V	-	-	12	mA DC
<b>CAN Specifications</b> (not available in no-CAN variants (*-NC))						
+VCAN	Input Voltage	-	5.5	12	-	V DC
	Input Current	+VCAN = 12 V DC	-	52	73	mA DC
R <sub>term</sub>	Termination resistance tolerance	-	118.8	120	121.2	Ω
	Power rating	-	-	-	0.125	W
CAN_P	Dominant Output	-	2.9	3.5	4.5	V DC
	Recessive Output	-	2	2.3	3	V DC
	Output Current	-	10	-	70	mA DC
	Output Signal Rise Time	-	-	20	50	ns
	Output Signal Fall Time	-	-	20	50	ns
CAN_N	Dominant Output	-	0.8	1.2	1.5	V DC
	Recessive Output	-	2	2.3	3	V DC
	Output Current	-	10	-	70	mA DC
	Output Signal Rise Time	-	-	20	50	ns
	Output Signal Fall Time	-	-	20	50	ns
Isolation	Rated Isolation	-	-	-	60	V
<b>RS-485 Modbus-RTU Specifications</b>						
+VMOD	Output Voltage	-	-	+VSYS	-	V DC
	Output Current	+VMOD = 24 V DC	-	-	1	A DC
R <sub>term</sub>	Termination resistance tolerance	-	148.5	150	151.5	Ω
	Power rating	-	-	-	0.125	W
V <sub>od</sub>	Driver differential output	-	1.5	2	-	V

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I <sub>o</sub>	Output current	-	-60	-	60	mA
t <sub>r</sub>	Output Signal Rise Time	-	0.3	0.7	1.2	μs
t <sub>f</sub>	Output Signal Fall Time	-	0.3	0.7	1.2	μs

## Environmental Conditions

Symbol	Parameter	Min	Typ	Max	Units
<b>Thermal Specifications</b>					
T <sub>a</sub>	Operating Temperature	-10	25	60	°C
	Storage Temperature	-20	25	60	°C
<b>Humidity Specifications</b>					
RH	Operational RH	5	-	85	%
	Storage RH	5	-	85	%
<b>Shock and Vibration Specifications</b>					
Vertical	Vertical shock/vibration	-	-	10	m/s <sup>2</sup>
Longitudinal	Longitudinal shock/vibration	-	-	10	m/s <sup>2</sup>
Transverse	Transverse shock/vibration	-	-	10	m/s <sup>2</sup>
Pulse vibration	On each axis	-	-	245	m/s <sup>2</sup>

The Stack Controller meets industry standards CISPR 22 Class A and IEC/EN 61000-4-2 for EMC/EMI and ESD respectively. All components are EU RoHS / China RoHS compliant.

The Stack Controller has been designed to meet the requirements of SAE J2464 (shock) and SAE J2380 (random vibration).

## Standards and Certifications

The Stack Controller meets industry standards CISPR 22 Class A and IEC/EN 61000-4-2 for EMC/EMI and ESD respectively. All components are EU RoHS / China RoHS compliant.

Standard/Certification		
Stationary Battery Safety	UL Recognized	UL 1973 (file no. MH64071)
Functional Safety	UL Recognized	UL 991 (file no. MH64071) UL 1998 (file no. MH64071)

UL 1973 recognition ensures safe battery operation and significantly reduce the effort of certifying the energy storage solution to meet UL 1973 and UL 9540.



## High-Voltage Power Interface



Exceeding the maximum ratings will damage the Power Interface module.

## Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
<b>External Power Specifications</b>						
+VIN	Input DC Voltage	-	13	24	34	V DC
	Input AC Voltage	-	16	20	24	V AC
	Input DC Current	+VIN = 24 V DC	-	-	3.5	A DC
	Input AC Current	+VIN = 24 V AC	-	-	5.5	A AC
	Input Isolation from Chassis/COM	-	60	-	-	V RMS
<b>Stack Power Specifications</b>						
+VSPIN	Input DC Voltage	-	0	-	1250	V DC
	Input DC Current	+VSPIN = 1250 V DC	-	-	313	μA DC
Vins	Internal reinforced insulation rating from Chassis/COM	-	-	-	1250	V DC
<b>Stack Bus Specifications</b>						
+VSYS	Output Voltage	-	13	24	34	V DC
	Output Current	+VSYS = 24 V DC	-	-	1.3	A DC
Rterm	Termination resistance tolerance	-	118.8	120	121.2	Ω
	Power rating	-	-	-	0.125	W
StackbusP	Dominant Output	-	2.45	-	3.3	V DC
	Recessive Output	-	-	2.3	-	V DC
	Output Current	-	10	-	50	mA DC
	Output Signal Rise Time	-	35	-	135	ns
	Output Signal Fall Time	-	35	-	135	ns
StackbusN	Dominant Output	-	0.5	-	1.25	V DC
	Recessive Output	-	-	2.3	-	V DC
	Output Current	-	10	-	50	mA DC
	Output Signal Rise Time	-	35	-	135	ns
	Output Signal Fall Time	-	35	-	135	ns
<b>Contactors Specifications</b>						

Symbol	Parameter	Conditions	Min	Typ	Max	Units
+VCOIL	External Coil Power Supply Input	-	5	24	40	V DC
	External Coil Power Supply Continuous Current	+VCOIL = 24 V DC	-	-	2.8	A DC
	External Coil Power Supply Pulse Current (<300 $\mu$ s)	+VCOIL = 24 V DC	-	-	20	A DC
+VINT	Internal Coil Power Supply Voltage	-	-	+VSYS	-	V DC
	Internal Coil Power Supply Current	-	-	-	2.8	A DC
COIL(n)	Coil Driver Output Voltage	-	-	+VCOIL	-	V DC
	Coil Driver Output Current	+VCOIL = 24 V DC	-	-	2.8	A DC
<b>Interlock Specifications</b>						
OVERRIDE	OverRide Voltage Output	+VCOIL = 24 V DC	-	5	-	V DC
	OverRide Current Output	+VCOIL = 24 V DC	49.5	50	50.5	mA DC
DRV	Drv Voltage Output	+VCOIL = 24 V DC	-	5	-	V DC
	Drv Current Output	+VCOIL = 24 V DC	49.5	50	50.5	mA DC
<b>Current Shunt Specifications</b>						
VSHUNT_REF	Reference Output Voltage	-	-	1.25	-	V DC
	Reference Output Current	-	-250	0	250	$\mu$ A DC
Vdiff	Differential voltage between VSHUNT_BAT and VSHUNT_LOAD	-	-1	0	1	V DC
Vmes	Measurement resolution	-	-	143	-	nV DC
Vins	Internal reinforced insulation rating from Chassis/COM	-	-	-	1250	V DC
<b>Thermistor Specifications</b>						
+VTHERM	Thermistor Output Voltage	-	-	2.5	-	V DC
	Thermistor Output Current	+VTHERM = 2.5 V DC	-	-	250	$\mu$ A
Rt	Thermistor Resistance at 25 °C	-	-	10	-	k $\Omega$
Vins	Internal reinforced insulation rating from Chassis/COM	-	-	-	1250	V DC



While High-Voltage BMS is UL-approved for use with +VSYS voltages down to 13 V DC, operation beyond the certified range is possible down to 10.6 V DC.

## Environmental Conditions

Symbol	Parameter	Min	Typ	Max	Units
<b>Thermal Specifications</b>					
Ta	Operating Temperature	-10	25	60	°C
	Storage Temperature	-20	25	60	°C
<b>Humidity Specifications</b>					
RH	Operational RH	5	-	85	%
	Storage RH	5	-	85	%
<b>Shock and Vibration Specifications</b>					
Vertical	Vertical shock/vibration	-	-	10	m/s <sup>2</sup>
Longitudinal	Longitudinal shock/vibration	-	-	10	m/s <sup>2</sup>
Transverse	Transverse shock/vibration	-	-	10	m/s <sup>2</sup>
Pulse vibration	On each axis	-	-	245	m/s <sup>2</sup>

The Power Interface has been designed to meet the requirements of SAE J2464 (shock) and SAE J2380 (random vibration).

## Standards and Certifications

The Power Interface meets industry standards CISPR 22 Class A and IEC/EN 61000-4-2 for EMC/EMI and ESD respectively. It has been designed to meet EN 60950 high voltage creepage/clearance distances for reinforced insulation rated to 1250 V DC. All components are EU RoHS / China RoHS compliant.

Standard/Certification		
Stationary Battery Safety	UL Recognized	UL 1973 (file no. MH64071)
Functional Safety	UL Recognized	UL 991 (file no. MH64071) UL 1998 (file no. MH64071)

UL 1973 recognition ensures safe battery operation and significantly reduce the effort of certifying the energy storage solution to meet UL 1973 and UL 9540.

## Cell Interface



Exceeding the maximum ratings will damage the Cell Interface module.

## Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
<b>Link In Specifications</b>						
$+V_{BUS}$	Input Voltage	-	9	24	60	V DC
	Input Current, CI-12	+VBUS = 24 V DC, Link Out disconnected	-	-	25.5	mA DC
	Input Current, CI-16 and CI-4M12	+VBUS = 24 V DC, Link Out disconnected	-	-	31.7	mA DC
$I_{P\_LINK}$	Output Current	-	-	-	20	mA DC
$I_{N\_LINK}$	Output Current	-	-	-	20	mA DC
<b>Link Out Specifications</b>						
$+V_{BUS}$	Output Voltage	-	-	+VBUS	-	V DC
	Output Current per CI-12	+VBUS = 24 V DC	-	-	25.5	mA DC
	Output Current per CI-16 and CI-4M12	+VBUS = 24 V DC	-	-	31.7	mA DC
$I_{P\_LINK}$	Output Current	-	-	-	20	mA DC
$I_{N\_LINK}$	Output Current	-	-	-	20	mA DC
<b>Battery Cells Specifications</b>						
$C_{(n)} - C_{(n-1)}$	Input Cell Voltage Range	CI-12, CI-16	0	-	5	V DC
$B_{(n)} - B_{(n-1)}$	Input Block Voltage Range	CI-4M12	5	-	20	V DC
$V_{sum}$	Voltage between C0 and C12	CI-12, +VBUS = 0 V DC	11	-	60	V DC
	Voltage between C0 and C8	CI-16, +VBUS = 0 V DC	11	-	40	V DC
	Voltage between C8 and C16	CI-16, +VBUS = 0 V DC	11	-	40	V DC
	Voltage between B0 and B2	CI-4M12, +VBUS = 0 V DC	11	-	40	V DC
	Voltage between B2 and B4	CI-4M12, +VBUS = 0 V DC	11	-	40	V DC
TME	Total Measurement Error	CI-12, CI-16, +VBUS = 24 V DC	±0.1	±1.2	±1.6	mV DC
	Total Measurement Error	CI-4M12, +VBUS = 24 V DC	±2.0	±8.0	±10.0	mV DC
$I_{(n)}$	Cell Balancing Current (only for CI-12 and CI-16)	$C_{(n)} - C_{(n-1)} = 4$ V DC	304	307	310	mA DC
$V_{bal}$	Cell Voltage for Balancing	CI-12 and CI-16	1.1	-	-	V DC

Symbol	Parameter	Conditions	Min	Typ	Max	Units
Vins	Internal reinforced insulation rating from Chassis/COM	-	-	-	1250	V DC
<b>Temperature Sensors Specifications</b>						
I <sub>(n)</sub>	Output Current to Temperature Sensor	-	-	-	300	μA
R <sub>t(n)</sub>	Temperature Sensor Resistance at 25 °C	-	-	10	-	kΩ
T <sub>(n)</sub>	Input Temperature Sensor Voltage Range	Cell 0 or Block 0 = 0 V	0	-	3	V
Vins	Internal reinforced insulation rating from Chassis/COM	-	-	-	1250	V DC

## Environmental Conditions

Symbol	Parameter	Min	Typ	Max	Units
<b>Thermal Specifications</b>					
T <sub>a</sub>	Operating Temperature	-10	25	60	°C
	Storage Temperature	-20	25	60	°C
<b>Humidity Specifications</b>					
RH	Operational RH	5	-	85	%
	Storage RH	5	-	85	%
<b>Shock and Vibration Specifications</b>					
Vertical	Vertical shock/vibration	-	-	10	m/s <sup>2</sup>
Longitudinal	Longitudinal shock/vibration	-	-	10	m/s <sup>2</sup>
Transverse	Transverse shock/vibration	-	-	10	m/s <sup>2</sup>
Pulse vibration	On each axis	-	-	245	m/s <sup>2</sup>

## Standards and Certifications

The Cell Interface meets industry standards CISPR 22 Class A and IEC/EN 61000-4-2 for EMC/EMI and ESD respectively. It has been designed to meet EN 60950 high voltage creepage/clearance distances for reinforced insulation rated to 1250 V DC. All components are EU RoHS / China RoHS compliant.

Certification/Report	
Stationary Battery Safety	UL Recognized - UL 1973 (file no. MH64071)
Functional Safety	- UL Recognized - UL 991 (file no. MH64071) - UL Recognized - UL 1998 (file no. MH64071)
Electrical Safety	IEC 62368-1:2014 (Second Edition), IEC 62368-3:2017
Industrial Immunity	EN/IEC 61000-6-2 2019

## Certification/Report

Shock and Vibration

Designed to meet the requirements of SAE J2464 and SAE J2380

UL 1973 recognition ensures safe battery operation and significantly reduce the effort of certifying the energy storage solution to meet UL 1973 and UL 9540.

## Maximum Stack Deployment

Cell Interface modules are deployed as a daisy chain to monitor the cells of a stack. The maximum number of modules that are supported in a stack depend on two metrics:

- the maximum number of modules that can be powered over Link Bus power (if required)
- the required scan rate of the cell voltage measurements

## Limits Due to Link Bus Power

Max CI-12	Max CI-16	Max CI-4M12
50	40	40

## Limits Due to Cell Voltage Scan Rate

The following are approximate cell voltage scan rates for different lengths of Cell Interface daisy chains where all cells are installed. They are provided for reference only and can vary depending by functionality enabled on the Nuvation Energy BMS.

**Table 68. Cell Voltage Scan Rates for CI-16 and CI-4M12**

Measurement Anti-Aliasing Filter	Cell Interface Chain Length	Scan Rate [Hz]
Off	1	5.53
Off	5	3.32
Off	10	2.22
Off	15	1.74
Off	20	1.38
Off	25	1.15
Off	30	0.91
On	1	1.55
On	5	1.31
On	10	1.11
On	15	0.98
On	20	0.84
On	25	0.76
On	30	0.63

**Table 69. Cell Voltage Scan Rates for CI-12**

Measurement Anti-Aliasing Filter	Cell Interface Chain Length	Scan Rate [Hz]
Off	1	6.01
Off	5	3.94
Off	10	3.03
Off	15	2.59
Off	20	1.97
Off	25	1.63
Off	30	1.44
Off	35	1.30
Off	40	1.08
On	1	1.58
On	5	1.38
On	10	1.26
On	15	1.14
On	20	1.00
On	25	0.94
On	30	0.86
On	35	0.80
On	40	0.72

## Appendix B: Ordering Information



High-Voltage BMS kits—which include the modules, cables, and power supply—are available to get you started quickly. Please visit <https://nstore.nuvationenergy.com> for more details.

**Table 70. High-Voltage BMS Ordering Information (kits)**

Part Number	Product Name
NUV100-BASE-NC-12-KIT	High-Voltage BMS Kit - 12 channel, no CAN
NUV100-BASE-NC-16-KIT	High-Voltage BMS Kit - 16 channel, no CAN
NUV100-BASE-NC-4M12-KIT	High-Voltage BMS Kit - 12V 4 channel, no CAN
<b>Available as special order</b>	
NUV100-BASE-12-KIT	High-Voltage BMS Kit - 12 channel
NUV100-BASE-16-KIT	High-Voltage BMS Kit - 16 channel
NUV100-BASE-4M12-KIT	High-Voltage BMS Kit - 12V 4 channel

## High-Voltage Stack Controller

Product part numbers for ordering a Stack Controller are listed in [Table 71, “Stack Controller Ordering Information”](#).

**Table 71. Stack Controller Ordering Information**

Part Number	Product Name
NUV100-SC-NC	High-Voltage Stack Controller, no CAN, DIN Mount
NUV100-SC-NC-U	High-Voltage Stack Controller, no CAN, PCB assembly only (no enclosure)
<b>Available as special order</b>	
NUV100-SC	High-Voltage Stack Controller, DIN Mount
NUV100-SC-U	High-Voltage Stack Controller, PCB assembly only (no enclosure)

## High-Voltage Power Interface

Product part numbers for ordering a Power Interface are listed in [Table 72, “Power Interface Ordering Information”](#).

**Table 72. Power Interface Ordering Information**

Part Number	Product Name
NUV100-PI-HE	High-Voltage Power Interface, DIN Mount



Part Number	Product Name
NUV100-PI-HE-U	High-Voltage Power Interface, PCB assembly only (no enclosure)



If mounting a Power Interface, PCB assembly only (no enclosure), note that the Power Interface contains high-voltage signals reaching as high as 1250 V DC. Care must be taken when mounting the PCB into a metal enclosure to ensure that the metal walls remain a safe distance from the exposed conductor on the PCB. Using 1250 V DC as an example, the metal walls of the enclosure must be at least 4.2 mm from the nearest exposed conductor and must not touch the PCB or any component on the PCB, including the connector housings.

## Cell Interface

Product part numbers for ordering a Cell Interface are listed in [Table 73, "Cell Interface Ordering Information"](#). Accessory kits are listed in [Table 74, "Cell Interface Accessory Kits Ordering Information"](#).



Cell Interface kits—which include the Cell Interface module and cables—are available to get you started quickly. Please visit <https://nstore.nuvationenergy.com> for more details.

**Table 73. Cell Interface Ordering Information**

Part Number	Product Name
NUV100-CI-12-1	Cell Interface - 12 channel, Bulkhead
NUV100-CI-12-U	Cell Interface - 12 channel, PCB assembly only (no enclosure)
NUV100-CI-12-KIT	Cell Interface Kit - 12 channel
NUV100-CI-16-1	Cell Interface - 16 channel, Bulkhead
NUV100-CI-16-U	Cell Interface - 16 channel, PCB assembly only (no enclosure)
NUV100-CI-16-KIT	Cell Interface Kit - 16 channel
NUV100-CI-4M12-1	Cell Interface - 12 V 4 channel, Bulkhead
NUV100-CI-4M12-U	Cell Interface - 12 V 4 channel, PCB assembly only (no enclosure)
NUV100-CI-4M12-KIT	Cell Interface Kit - 12V 4 channel



If mounting a Cell Interface, PCB assembly only (no enclosure), note that the Cell Interface contains high-voltage signals reaching as high as 1250 V DC. Care must be taken when mounting the PCB into a metal enclosure to ensure that the metal walls remain a safe distance from the exposed conductor on the PCB. Using 1250 V DC as an example, the metal walls of the enclosure must be at least 4.2 mm from the nearest exposed conductor and must not touch the PCB or any component on the PCB, including the connector housings.

**Table 74. Cell Interface Accessory Kits Ordering Information**

Part Number	Product Name
NUVP-CI-DIN-MB	Cell Interface Mounting Bracket (Bulkhead-to-DIN)

## Appendix C: Changing the Nuvation Energy BMS Network Configuration

Nuvation Energy BMS is flexible and allows changing its network configuration if you need the IP address of Nuvation Energy BMS to match your existing network settings.



Improperly modifying the network settings can result in a Nuvation Energy BMS that cannot communicate over Ethernet.

To restore Nuvation Energy BMS to its default IP address (192.168.1.21), you must initiate a factory reset action—refer to the [Appendix D](#) for instructions.

Be sure to record the updated IP address, netmask, and gateway settings if you decide to make changes to these settings.

### Modifying the Network Settings

1. Unlock the Operator Interface by clicking the menu and Unlock
  - You will need the Operator Interface password if the Operator Interface is password-protected
2. Follow the instructions in [Section 6.3.2.5, “Editing an IP Address”](#) to update the IP.

## Appendix D: Factory Reset

In the unlikely event High-Voltage BMS becomes inaccessible due to a forgotten password or invalid network configuration; a factory reset operation must be performed to restore the system to the default settings.

### Factory Reset Cable

To proceed you will need a 'Factory Reset' cable. This cable may be purchased by contacting [support@nuvationenergy.com](mailto:support@nuvationenergy.com).

Alternatively, you may connect an external 18–24 V DC power supply to the Stack Controller module's Link Bus connector. Pin 1 on the Link Out is positive and Pin 2 is negative

### Reset High-Voltage BMS to factory defaults

1. Disconnect the External Power from the Power Interface
2. Disconnect the Link Bus from the Stack Controller module's Link Out connector
3. Connect the Factory Reset cable to the Stack Controller module's Link Out connector
4. Observe a blinking activity LED on the Stack Controller
5. Connect External Power to the Power Interface
6. Disconnect the Factory Reset cable from the Stack Controller
7. Wait for the blinking activity LED on the Stack Controller to stop blinking
8. Reconnect the Link Bus to the Stack Controller module's Link Out connector

High-Voltage BMS should now be reset to the factory firmware version, default password, and default network configuration.

## Appendix E: Best Practices

This section describes important concepts which need special attention to achieve a reliable installation.

### Grounding

It is assumed that Nuvation Energy BMS will be attached electrically to an earth or local chassis ground point, via the DIN rail grounding provision (#8-32 , ¼" Hex-head drive, earth grounding screw), and the mounting brackets on Nuvation Energy BMS component enclosures.

Voltages and signals on the Stack Bus and Link Bus cables are chassis/earth ground referenced. In addition, the Stack Controller module's USB port, non-isolated RS485, and GPIO-In signals; and the Power Interface module's Contactor coils and Interlock signals are chassis/earth ground referenced.

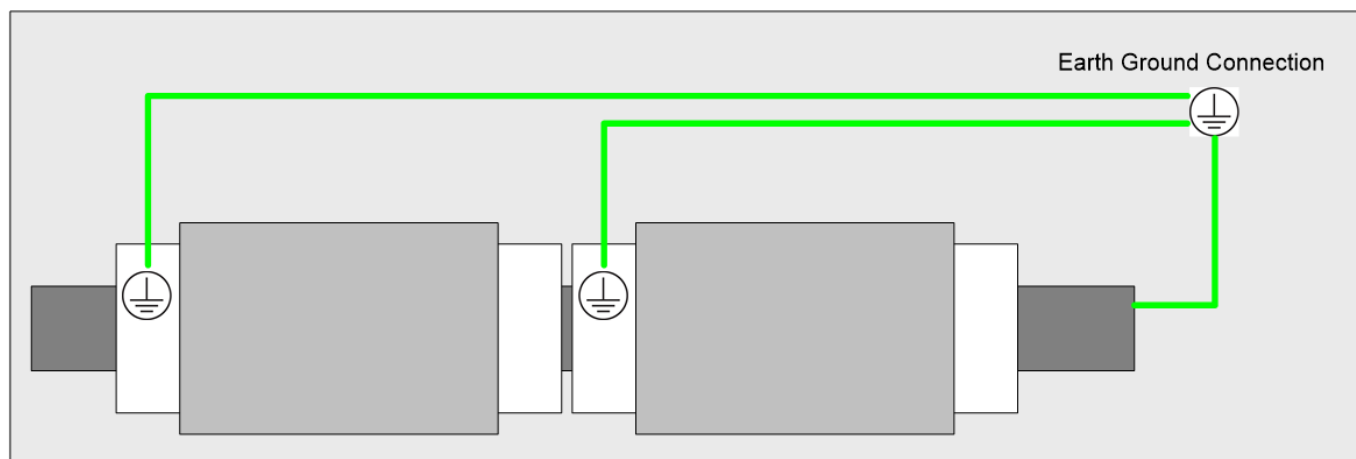
All connections to the battery stack are isolated from chassis ground. This includes:

- The Current Shunt, Thermistor, and Stack Power connections on the Power Interface
- The Battery Cells and Temperature Sensors connections on the Cell Interface
- The Ethernet and CAN interfaces on the Stack Controller

It is acceptable, as may be required in some cases, for the battery stack to be ground-referenced at some single point. However, a 24 V RMS AC or 24 V DC power supply connected to the Power Interface module's External Power connection must be isolated from earth/chassis ground, with a working isolation voltage of at least 60 V RMS for all Power Interface models.

Protective earthing conductors must be attached to each DIN enclosure at the designated ground screw location on the DIN clip. Furthermore, the DIN rail itself should be connected to earth ground. 14 AWG wire with a jacket color appropriate for indicating it is a protective earthing conductor must be used.

An example of this grounding scheme is shown below:



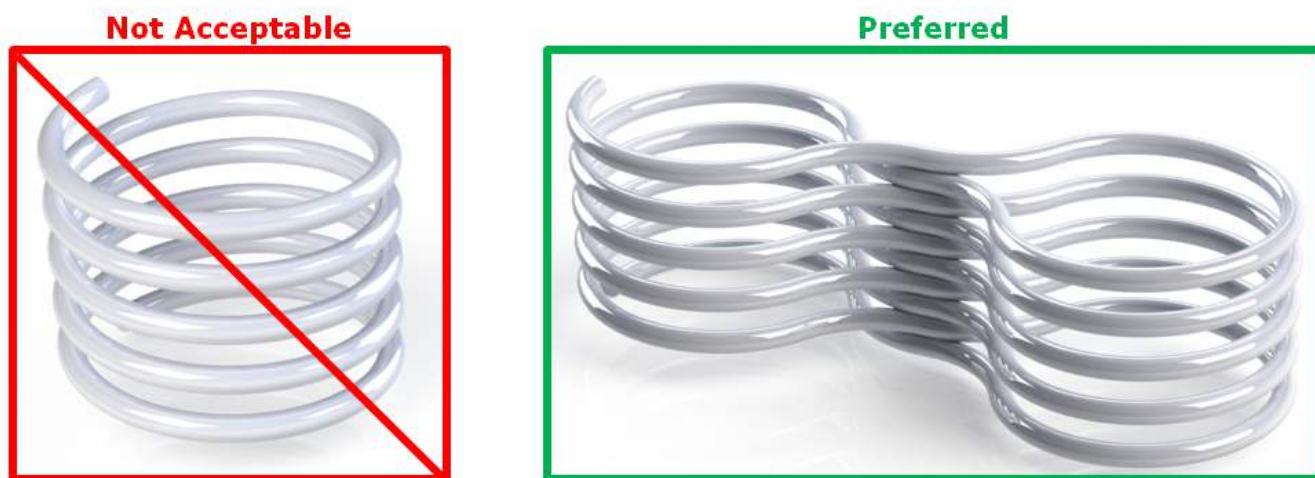
**Figure 80. Example Earth Ground Wiring Diagram**

## Excess Cable Management

During the first prototype system build, it is possible to encounter cable lengths that are too long for your system. Leaving the excess cable length unmanaged can result in a messy system installation.

If reducing the cable length is not feasible or if there is no time to physically modify the lengths, a common solution is to wrap the excess cable length in a coil and fasten the wire loop in the cabinet. This basic tactic has the undesirable effect of creating an air-core transformer which will couple EMI into the cable extremely well.

The best solution to cable length management is to bundle the excess length in a figure-8 pattern. This prevents the bundle from turning into an air-core transformer since the direction of current in one side of the figure-8 turns opposite to the current in the other side. It is recommended to use the figure-8 method if physically reducing the cable length is not possible.



**Figure 81. Excess Cable Management Examples**

## System Noise

High-power inverters generate a lot of system noise, especially on the DC bus. This is due to the industry standards for AC harmonics and EMC on the grid-side which require the DC bus to help filter out the harmful emissions. Unfortunately, that means the battery cells and the High-Voltage BMS experience extreme levels of noise.

The most harmful emissions on the DC bus are between the DC bus and earth. This is due to the slew-rate of the switching devices implemented in the inverter (usually IGBTs). The slew-rate is impacted by many elements, and the emissions can be minimized by carefully grounding the installation so that the return-path for the high-frequency switching noise can be kept small.

The High-Voltage BMS has various faults and informative registers to determine if the system has a high level of noise that is impacting the Battery Management System.

The communication faults are:

```
sc_fault_linkbus_wdt.trig  
sc_fault_stackbus_rxwdt.trig  
sc_fault_stackbus_txwdt.trig  
sc_fault_pi_afe_wdt.trig  
sc_fault_controller_wdt.trig
```

The informative communication error registers are:

```
sc_linkbus_packets.operation_read_errors  
sc_linkbus_packets.operation_validate_errors  
sc_stackbus.rxerrrate  
sc_stackbus.txerrrate  
pi_afe.rx_err_rate  
pi_afe.tx_err_rate
```

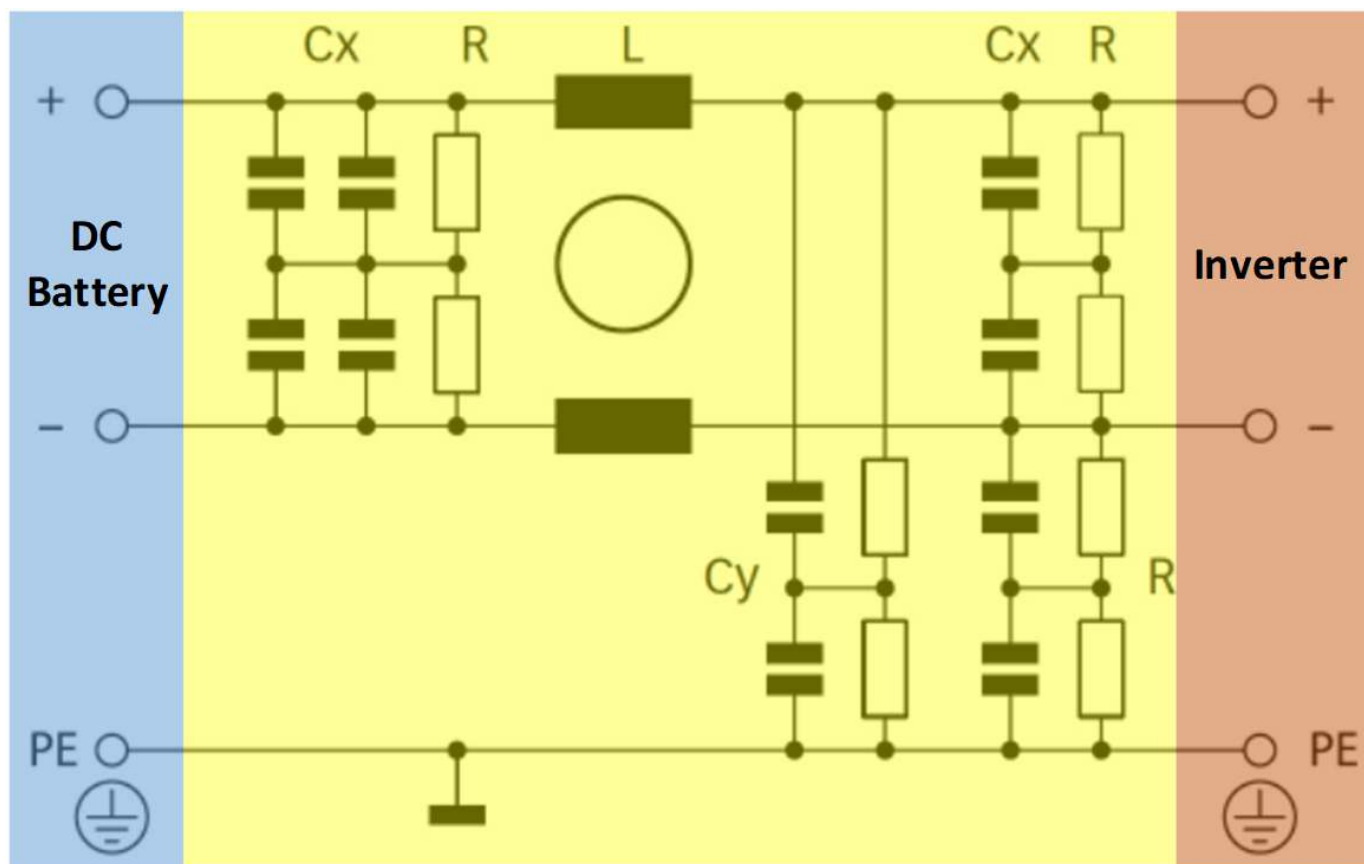
The system controller heartbeat should also be coming through as expected, and can be verified by reading the register:

```
sc_controller_heartbeat.value
```

In a correctly wired system, a communication fault points to elevated system noise that is disrupting communications. If the system grounding scheme cannot be improved, there are still a few techniques within High-Voltage BMS or the battery area to try to decrease the amount of noise.

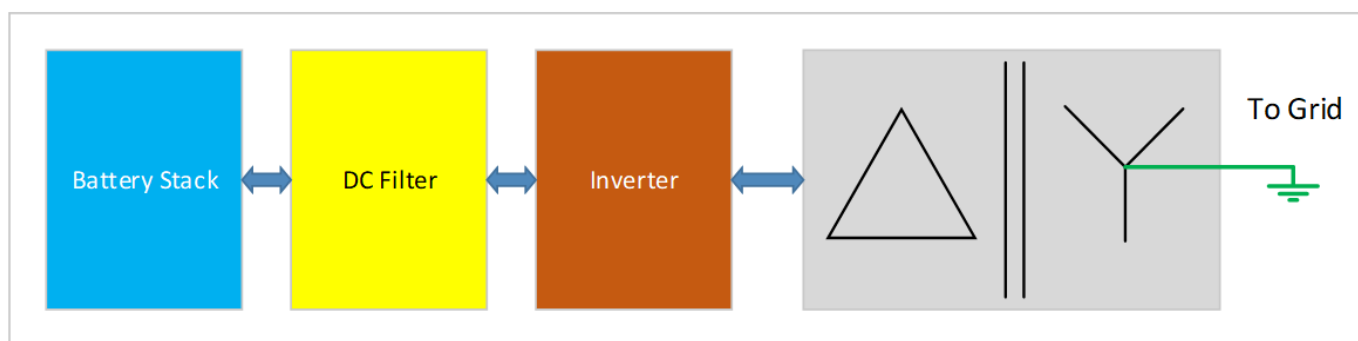
## DC Filtering

A DC filter can be installed between the DC bus and the inverter or between each DC battery stack and the common DC bus in a multi-stack system. Schaffner FN 2200 is an example DC filter which has been known to decrease the amount of harmful emissions on the DC bus. An example filter installation is shown below:



**Figure 82. Example DC Filter Schematic**

When using DC filters, please be aware that it shunts high-frequency noise to earth. If the inverter is not driving an insulated neutral system, there will be high current pulses flowing in the system earth which can trip ground fault detectors. It might be necessary to install an isolation transformer between the inverter and the grid to remove the high current pulses.



**Figure 83. Example Isolation Transformer Installation Diagram**

## Stack Controller Grounding

If `sc_stackbus` communication faults are occurring, it might be alleviated by providing a dedicated ground connection for the Stack Controller. In the typical installation, the Stack Controller is grounded through the Power Interface via the Stack Bus cable. By providing a direct connection to



ground, any noise entering the Stack Controller from the Link Bus cable will not flow through the Stack Bus cable.

Connect one of the following pins to earth:

- Link Out pin 2
- CAN485 pin 4
- GPIO-Out pin 5
- GPIO-In pin 1, 2, 3 or 4

## Link Bus Power

While the communication interface between the Stack Controller and the Cell Interface is a daisy-chain, the power supplied to the Cell Interface from the Stack Controller is a bus. This results in the power twisted pair in the Link Bus cable carrying power up the entire length of the chain. This provides a decent medium to couple system noise into the Link Bus which can result in `sc_linkbus` communication faults.

In systems where the cells can provide the necessary minimum operating voltage to the Cell Interface, Link Bus power can be disabled if the observed impact on performance is acceptable. The power twisted pair must be disconnected in the Link Bus cable, so all Link Out and Link In connectors on the Stack Controller and the Cell Interface modules must have pins 1 and 2 unpopulated.

Also, the High-Voltage BMS must be configured to disable power to the Link Bus, by setting this register to 0:

```
sc_linkbus.power_mode = 0
```

*From time to time Nuvation Energy will make updates to products in response to changes in available technologies, client requests, emerging energy storage standards, and other industry requirements. The product specifications in this document, therefore, are subject to change without notice.*

© 2023 Nuvation Energy

## Registers

### B

Breaker Conflict Check, [171](#)  
Breaker Tripped Check, [171](#)

### C

cell, [88](#)  
    installed, [88](#), [115](#), [116](#)  
    voltage, [115](#)  
Cell High Voltage, [165](#)  
Cell Interface Measurement Check, [169](#)  
Cell Interface Measurement Watchdog, [171](#)  
Cell Low Voltage, [165](#)  
Cell Open Wire Measurement Watchdog, [171](#)  
Cell Open Wire Test, [169](#), [169](#)  
Cell Over Voltage, [165](#)  
Cell Under Voltage, [165](#)  
Cell Voltage Measurement Watchdog, [171](#)  
Charge High Current, [167](#)  
Charge High Temperature, [167](#)  
Charge Low Temperature, [167](#)  
Charge Over Current, [167](#)  
Charge Over Temperature, [167](#)  
Charge Under Temperature, [167](#)  
Clock Check, [173](#)  
Contactor Coil Drive Check, [168](#)  
Contactor End-Of-Life, [170](#)  
Contactor End-Of-Life Watchdog, [172](#)  
Contactor Feedback Check, [168](#)  
Contactor Monitor Watchdog, [172](#)  
Controller Heartbeat Watchdog, [171](#), [171](#)  
Current Shunt Open Wire Check, [169](#)  
Current Shunt Short Circuit Check, [169](#)

### D

Discharge High Current, [167](#)  
Discharge High Temperature, [167](#)  
Discharge Over Current, [167](#)  
Discharge Under Temperature, [167](#), [167](#)

### F

Fault Pilot Check, [169](#)  
Firmware Version Mismatch Check, [173](#)

### I

Interlock Check, [168](#)

### L

LinkBus Communication Watchdog, [172](#)  
LinkBus Power Mode Check, [173](#)

### P

pi\_afe\_iadc  
    divider, [122](#)  
    multiplier, [122](#)  
pi\_afe\_vadc  
    divider, [124](#)  
    multiplier, [124](#)  
pi\_clock\_check  
    enabled, [129](#)  
pi\_fault\_config\_factory\_verify, [173](#)  
pi\_fault\_factory\_verify\_wdt, [172](#)  
Power Interface AFE Watchdog, [172](#)  
Power Interface Factory Memory Check, [173](#)  
Power Interface Factory Memory Watchdog, [172](#)  
Power Interface Measurement Watchdog, [171](#)  
Pre-charge Over Current, [168](#)  
Pre-charge Timeout, [168](#)

### S

sc\_canbus  
    base\_can\_address, [112](#), [112](#), [113](#), [113](#), [115](#),  
        [115](#), [116](#), [118](#)  
    enable, [112](#)  
    enabled, [112](#)  
    numtoread, [115](#)  
    report\_interval, [112](#), [112](#)  
    report\_msg\_interval, [112](#), [112](#), [118](#)  
sc\_canbus\_bulk\_map, [115](#), [115](#), [115](#)  
    baseaddress, [115](#), [115](#), [119](#), [119](#)  
    baseenableaddress, [115](#)  
    baseenabled, [119](#)  
    baseenabledaddress, [115](#), [119](#)  
    numtoread, [115](#), [115](#), [115](#), [116](#), [119](#)  
    offset, [115](#), [115](#)  
sc\_canbus\_map  
    address, [112](#), [113](#)  
sc\_canbus\_packets  
    err\_rate\_window, [112](#), [112](#)  
sc\_clock  
    seconds, [113](#)  
sc\_controller\_heartbeat  
    value, [113](#)  
sc\_controller\_wdt  
    period, [100](#)

[sc\\_fault\\_ci](#), [169](#), [171](#)  
[sc\\_fault\\_clocks](#), [173](#)  
     [disabled](#), [128](#)  
[sc\\_fault\\_config](#), [173](#)  
[sc\\_fault\\_config\\_factory](#), [173](#)  
[sc\\_fault\\_config\\_factory\\_verify](#), [173](#)  
[sc\\_fault\\_controller\\_wdt](#), [171](#)  
     [disabled](#), [100](#)  
[sc\\_fault\\_factory\\_verify\\_wdt](#), [172](#)  
[sc\\_fault\\_fault\\_pilot\\_state\\_mismatch](#), [169](#)  
[sc\\_fault\\_fw\\_mismatch](#), [173](#)  
[sc\\_fault\\_linkbus\\_power](#), [173](#)  
[sc\\_fault\\_linkbus\\_wdt](#), [172](#)  
[sc\\_fault\\_pi\\_afe\\_wdt](#), [172](#)  
[sc\\_fault\\_pi\\_interlock](#), [168](#)  
[sc\\_fault\\_ram\\_test\\_wdt](#), [172](#)  
[sc\\_fault\\_register\\_link](#), [174](#)  
[sc\\_fault\\_rom\\_test\\_wdt](#), [172](#)  
[sc\\_fault\\_stackbus\\_rxwdt](#), [171](#)  
[sc\\_fault\\_stackbus\\_txwdt](#), [172](#)  
[sc\\_gpi](#)  
     [address](#), [111](#), [111](#)  
     [address\\_enabled](#), [110](#), [119](#)  
     [falling\\_edge\\_triggered](#), [111](#)  
     [inverted](#), [110](#), [111](#), [119](#)  
     [rising\\_edge\\_triggered](#), [111](#), [111](#)  
     [value](#), [119](#), [119](#)  
[sc\\_gpo](#)  
     [address](#), [110](#), [110](#)  
     [address\\_enabled](#), [110](#)  
     [inverted](#), [110](#), [110](#)  
[sc\\_gpo\\_fault\\_pilot](#)  
     [fault\\_delay](#), [128](#)  
     [fault\\_enabled](#), [128](#)  
[sc\\_linkbus](#)  
     [balance\\_settle\\_period](#), [125](#)  
     [cell\\_open\\_wire\\_bal\\_tol](#), [127](#)  
     [cell\\_open\\_wire\\_disabled](#), [126](#)  
     [cell\\_open\\_wire\\_low\\_thresh](#), [128](#)  
     [cell\\_open\\_wire\\_max](#), [127](#)  
     [cell\\_open\\_wire\\_period](#), [126](#)  
     [cell\\_open\\_wire\\_ratio\\_lower](#), [127](#)  
     [cell\\_open\\_wire\\_ratio\\_upper](#), [127](#)  
     [cell\\_open\\_wire\\_thresh](#), [127](#)  
     [cell\\_voltage\\_lpf\\_bw](#), [126](#)  
     [cicount](#), [125](#)  
     [power\\_mode](#), [125](#)  
     [scan\\_period](#), [125](#)  
     [softwareid](#), [125](#)  
[sc\\_modbus\\_rtu](#)  
     [device\\_address](#), [119](#)  
  
[sc\\_ram\\_test\\_wdt](#), [135](#)  
[sc\\_rom\\_test\\_wdt](#), [135](#)  
[sc\\_warn\\_controller\\_wdt](#), [171](#)  
[ssg\\_contactor\\_life\\_tracker](#)  
     [current\\_capture\\_window](#), [134](#)  
     [enabled](#), [133](#)  
     [is\\_precharge](#), [133](#)  
     [max\\_life](#), [134](#)  
     [max\\_opening\\_current](#), [134](#)  
     [max\\_rated\\_current](#), [134](#)  
     [min\\_opening\\_current](#), [134](#)  
[ssg\\_contactor\\_monitor](#)  
     [enabled](#), [133](#)  
[ssg\\_fault\\_config\\_contactor\\_life](#), [170](#)  
[ssg\\_fault\\_config\\_contactor\\_life\\_verify](#), [170](#)  
[ssg\\_fault\\_contactor\\_end\\_of\\_life](#), [170](#)  
     [thresh](#), [134](#)  
[ssg\\_fault\\_contactor\\_end\\_of\\_life\\_wdt](#), [172](#)  
[ssg\\_fault\\_contactor\\_life\\_sanity\\_check](#), [170](#)  
[ssg\\_fault\\_contactor\\_life\\_verify\\_wdt](#), [172](#)  
[ssg\\_fault\\_contactor\\_monitor\\_wdt](#), [172](#)  
[ssg\\_warn\\_contactor\\_end\\_of\\_life](#)  
     [thresh](#), [134](#)  
[Stack Controller Configuration Consistency Check](#), [173](#)  
[Stack Controller Factory Consistency Check](#), [173](#)  
[Stack Controller Factory Memory Check](#), [173](#)  
[Stack Controller Factory Memory Watchdog](#), [172](#)  
[Stack Controller RAM Test Watchdog](#), [172](#)  
[Stack Controller ROM Test Watchdog](#), [172](#)  
[Stack High Voltage](#), [166](#)  
[Stack Low Voltage](#), [166](#)  
[Stack Over Voltage](#), [166](#)  
[Stack Switchgear Contactor Life Memory Check](#), [170](#)  
[Stack Switchgear Contactor Life Memory Watchdog](#), [172](#)  
[Stack Switchgear Contactor Life Sanity Check](#), [170](#), [170](#)  
[Stack Under Voltage](#), [166](#)  
[stack\\_breaker](#)  
     [enable](#), [107](#)  
     [tripped\\_address](#), [107](#)  
     [tripped\\_address\\_enable](#), [107](#)  
[stack\\_cell\\_balancer](#)  
     [enabled](#), [106](#)  
     [max\\_enable\\_current](#), [107](#)  
     [max\\_enable\\_temperature](#), [106](#)  
     [min\\_enable\\_current](#), [107](#)  
     [min\\_enable\\_voltage](#), [106](#), [106](#)  
     [voltage\\_delta](#), [106](#), [106](#)

stack_cell_stat	temperature_discharge_low, <a href="#">95</a>
avg, <a href="#">113</a>	temperature_discharge_max, <a href="#">95</a> , <a href="#">95</a>
max, <a href="#">113</a>	temperature_discharge_min, <a href="#">95</a> , <a href="#">95</a>
min, <a href="#">113</a>	voltage_cell_high, <a href="#">92</a> , <a href="#">106</a>
stack_charge_status	voltage_cell_low, <a href="#">92</a>
hold_current, <a href="#">78</a> , <a href="#">123</a>	voltage_cell_max, <a href="#">92</a> , <a href="#">92</a> , <a href="#">99</a>
stack_contactor	voltage_cell_min, <a href="#">92</a> , <a href="#">92</a>
address, <a href="#">109</a>	voltage_stack_high, <a href="#">99</a>
address_enabled, <a href="#">108</a>	voltage_stack_low, <a href="#">99</a>
coil_error, <a href="#">109</a>	voltage_stack_max, <a href="#">99</a>
delay, <a href="#">109</a>	voltage_stack_min, <a href="#">99</a> , <a href="#">99</a>
direction, <a href="#">109</a>	stack_fault_breaker_conflict, <a href="#">171</a>
feedback_enable, <a href="#">109</a>	stack_fault_breaker_tripped, <a href="#">171</a>
feedback_error, <a href="#">109</a> , <a href="#">109</a>	disabled, <a href="#">108</a>
feedback_value, <a href="#">109</a> , <a href="#">111</a>	stack_fault_cell_hi, <a href="#">165</a>
installed, <a href="#">109</a>	thresh, <a href="#">92</a>
inverted, <a href="#">109</a>	stack_fault_cell_lo, <a href="#">165</a>
value, <a href="#">109</a>	thresh, <a href="#">92</a>
stack_control	stack_fault_cell_open_wire, <a href="#">169</a> , <a href="#">171</a>
auto_connect, <a href="#">102</a>	stack_fault_cell_over, <a href="#">165</a>
connect_delay, <a href="#">103</a> , <a href="#">103</a>	thresh, <a href="#">92</a>
connection_state, <a href="#">113</a>	stack_fault_cell_under, <a href="#">165</a>
disable_connecting, <a href="#">102</a>	thresh, <a href="#">93</a>
disable_precharge, <a href="#">102</a>	stack_fault_cell_wdt, <a href="#">171</a>
disconnect_delay, <a href="#">104</a>	stack_fault_charge_current_hi, <a href="#">167</a>
main_switch_state, <a href="#">109</a>	thresh, <a href="#">97</a>
precharge_bias_current, <a href="#">78</a>	stack_fault_charge_current_over, <a href="#">167</a>
precharge_bias_delay, <a href="#">102</a>	thresh, <a href="#">97</a>
precharge_delay, <a href="#">102</a> , <a href="#">103</a>	stack_fault_charge_therm_hi, <a href="#">167</a>
precharge_max_current, <a href="#">78</a> , <a href="#">103</a>	thresh, <a href="#">94</a>
precharge_switch_state, <a href="#">109</a>	stack_fault_charge_therm_lo, <a href="#">167</a>
requested_state, <a href="#">111</a> , <a href="#">113</a>	thresh, <a href="#">95</a>
stack_switch_state, <a href="#">109</a>	stack_fault_charge_therm_over, <a href="#">167</a>
stack_current_limit	thresh, <a href="#">94</a>
attack_settling_time, <a href="#">105</a> , <a href="#">105</a>	stack_fault_charge_therm_under, <a href="#">167</a>
charge_current_disable, <a href="#">110</a> , <a href="#">113</a>	thresh, <a href="#">95</a>
charge_current_limit, <a href="#">113</a>	stack_fault_ci_therm_consistency, <a href="#">169</a> , <a href="#">171</a>
charge_current_percent, <a href="#">113</a>	thresh, <a href="#">131</a>
decay_settling_time, <a href="#">105</a>	stack_fault_ci_wdt, <a href="#">171</a>
discharge_current_disable, <a href="#">110</a> , <a href="#">113</a>	stack_fault_coil_fail, <a href="#">168</a>
discharge_current_limit, <a href="#">113</a>	stack_fault_contactor_feedback_fail, <a href="#">168</a>
discharge_current_percent, <a href="#">113</a>	stack_fault_discharge_current_hi, <a href="#">167</a>
max_charge_current, <a href="#">78</a> , <a href="#">78</a> , <a href="#">104</a>	thresh, <a href="#">97</a>
max_discharge_current, <a href="#">104</a>	stack_fault_discharge_current_over, <a href="#">167</a>
min_charge_current, <a href="#">78</a> , <a href="#">104</a>	thresh, <a href="#">97</a>
nonlinear_rolloff, <a href="#">105</a>	stack_fault_discharge_therm_hi, <a href="#">167</a>
temperature_charge_high, <a href="#">95</a>	thresh, <a href="#">95</a>
temperature_charge_low, <a href="#">95</a>	stack_fault_discharge_therm_lo, <a href="#">167</a>
temperature_charge_max, <a href="#">94</a> , <a href="#">95</a>	thresh, <a href="#">95</a>
temperature_charge_min, <a href="#">95</a> , <a href="#">95</a>	stack_fault_discharge_therm_over, <a href="#">167</a>
temperature_discharge_high, <a href="#">95</a>	thresh, <a href="#">95</a>

stack\_fault\_discharge\_therm\_under, [167](#)  
     thresh, [95](#)  
 stack\_fault\_open\_shunt, [169](#), [171](#)  
     thresh, [129](#)  
 stack\_fault\_open\_wire\_wdt, [171](#)  
 stack\_fault\_power\_wdt, [171](#)  
 stack\_fault\_precharge\_over\_current, [168](#)  
     thresh, [78](#), [103](#)  
 stack\_fault\_precharge\_timeout, [168](#)  
     disabled, [103](#)  
 stack\_fault\_short\_shunt, [169](#), [171](#)  
     thresh, [131](#)  
 stack\_fault\_therm\_circuit\_fail, [169](#)  
 stack\_fault\_therm\_wdt, [171](#)  
 stack\_fault\_voltage\_hi, [166](#)  
     thresh, [99](#)  
 stack\_fault\_voltage\_lo, [166](#)  
     thresh, [99](#)  
 stack\_fault\_voltage\_over, [166](#)  
     thresh, [99](#), [100](#)  
 stack\_fault\_voltage\_sum, [166](#)  
     thresh, [100](#)  
 stack\_fault\_voltage\_under, [166](#)  
     thresh, [99](#)  
 stack\_open\_shunt\_detector  
     current\_activity\_threshold, [129](#)  
     enabled, [129](#)  
     filter\_sample\_count, [129](#)  
     minimum\_resistance\_threshold, [129](#)  
 stack\_power  
     current, [113](#)  
     voltage, [113](#)  
 stack\_short\_shunt\_detector  
     activity\_detection\_time, [130](#), [170](#)  
     current\_activity\_threshold, [130](#)  
     current\_time\_constant, [130](#)  
     enabled, [130](#)  
     filter\_sample\_count, [131](#)  
     inactivity\_envelope, [170](#)  
     ramp\_detection\_time, [130](#)  
     voltage\_activity\_threshold, [130](#), [170](#)  
     voltage\_time\_constant, [130](#)  
 stack\_soc  
     dod, [113](#)  
     nominal\_capacity, [83](#)  
     nominal\_cycle\_count, [84](#)  
     soc, [113](#)  
     vempty, [85](#), [92](#)  
     vemptyavg, [85](#)  
     vfull, [85](#), [92](#)  
     vfullavg, [85](#)

stack\_therm\_poly  
     coeff0, [120](#)  
     coeff1, [120](#)  
     coeff2, [120](#)  
     coeff3, [120](#)  
     coeff4, [120](#)  
     coeff5, [120](#)  
     coeff6, [120](#)  
 stack\_therm\_stat  
     avg, [113](#)  
     max, [113](#)  
     min, [113](#)  
 stack\_trig\_cell\_hi  
     thresh, [93](#)  
 stack\_trig\_cell\_lo  
     thresh, [93](#)  
 stack\_trig\_charge\_current\_hi  
     thresh, [97](#)  
 stack\_trig\_charge\_therm\_hi  
     thresh, [96](#)  
 stack\_trig\_charge\_therm\_lo  
     thresh, [96](#)  
 stack\_trig\_discharge\_current\_hi  
     thresh, [97](#)  
 stack\_trig\_discharge\_therm\_hi  
     thresh, [96](#)  
 stack\_trig\_discharge\_therm\_lo  
     thresh, [96](#)  
 stack\_trig\_voltage\_hi  
     thresh, [99](#)  
 stack\_trig\_voltage\_lo  
     thresh, [99](#)  
 stack\_trigger\_summary  
     clear\_faults, [111](#), [113](#)  
     fault, [110](#)  
     no\_faults, [113](#)  
 stack\_uvlo\_cell\_voltage  
     thresh, [93](#), [128](#)  
 stack\_uvlo\_stack\_voltage  
     thresh, [99](#), [128](#)  
 stack\_warn\_cell\_hi  
     thresh, [92](#)  
 stack\_warn\_cell\_lo  
     thresh, [92](#)  
 stack\_warn\_cell\_open\_wire, [169](#)  
 stack\_warn\_charge\_current\_hi  
     thresh, [97](#)  
 stack\_warn\_charge\_therm\_hi  
     thresh, [94](#)  
 stack\_warn\_charge\_therm\_lo  
     thresh, [95](#)

stack\_warn\_discharge\_current\_hi  
thresh, [97](#)

stack\_warn\_discharge\_therm\_hi  
thresh, [95](#)

stack\_warn\_discharge\_therm\_lo  
thresh, [95](#)

stack\_warn\_voltage\_hi  
thresh, [99](#)

stack\_warn\_voltage\_lo  
thresh, [99](#)

StackBus Receive Watchdog, [171](#)

StackBus Transmit Watchdog, [172](#)

## T

Temperature Measurement Watchdog, [171](#)

therm, [88](#)  
installed, [88](#), [115](#), [116](#)  
temperature, [115](#)

Thermistor Circuit Test, [169](#)

Thermistor Consistency Check, [169](#)

trigger\_name  
disabled, [90](#), [90](#), [91](#)  
end\_time\_hyst, [91](#)  
latched, [91](#)  
thresh, [90](#), [91](#)  
time\_hyst, [91](#)

## V

Voltage Sum Check, [166](#)